Many Random Walks Are Faster Than One

Noga Alon
Tel Aviv University
nogaa@tau.ac.il

Chen Avin
Ben-Gurion University
avin@cse.bgu.ac.il

Michal Koucký*
Academy of Sciences of Czech Republic
koucky@math.cas.cz

Gady Kozma
Weizmann Institute of Science
gady.kozma@weizmann.ac.il

Zvi Lotker
Ben-Gurion University
zvilo@cse.bgu.ac.il

Mark R. Tuttle
Intel
tuttle@acm.org

ABSTRACT

We pose a new and intriguing question motivated by distributed computing regarding random walks on graphs: How long does it take for several independent random walks, starting from the same vertex, to cover an entire graph? We study the *cover time*–the expected time required to visit every node in a graph at least once–and we show that for a large collection of interesting graphs, running many random walks in parallel yields a speed-up in the cover time that is linear in the number of parallel walks. We demonstrate that an exponential speed-up is sometimes possible, but that some natural graphs allow only a logarithmic speed-up. A problem related to ours (in which the walks start from some probabilistic distribution on vertices) was previously studied in the context of space efficient algorithms for undirected s-t-connectivity and our results yield, in certain cases, an improvement upon some of the earlier bounds.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Probabilistic algorithms; F.2 [**Analysis of Algorithms and Problem Complexity**]: Miscellaneous

General Terms

Algorithms, Theory

Keywords

Random Walks, Cover Time, Speed-up, Distributed Algorithms, Graph Search

*Supported in part by grant GA ČR 201/07/P276, project No. 1M0021620808 of MŠMT ČR and Institutional Research Plan No. AV0Z10190503.

SPAA'08, June 14–16, 2008, Munich, Germany.

1. INTRODUCTION

Consider the problem of hunting or tracking on a graph. The prey begins at one node, the hunters begin at other nodes, and in every step each player can traverse an edge of the graph. The goal is for the hunters to locate and track the prey as quickly as possible. What is the best algorithm for the hunters to explore the graph and find the prey? The answer depends on many factors, such as the nature of the graph, whether the graph can change dynamically, how much is known about the graph, and how well the hunters can communicate and coordinate their actions. Graph exploration problems such as this are particularly interesting in changing or unknown environments. In such environments, randomized algorithms are at an advantage, since they typically require no knowledge of the graph topology.

Random walks are a natural and thoroughly studied approach to randomized graph exploration. A *simple random walk* is a stochastic process that starts at one node of a graph, and at each step moves from the current node to an adjacent node chosen randomly and uniformly from the neighbors of the current node. A natural example of a random walk in a communication network arises when messages are sent at random from device to device. Since such algorithms exhibit locality, simplicity, low-overhead, and robustness to changes in the graph structure, applications based on random walks are becoming more and more popular. In recent years, random walks have been proposed in the context of querying, searching, routing, and self-stabilization in wireless ad-hoc networks, peer-to-peer networks, and other distributed systems and applications [17, 31, 12, 30, 8, 21, 1, 10].

The problem with random walks, however, is latency. In the case of a ring, for example, a random walk requires an expected $\Theta(n^2)$ steps to traverse a ring, whereas a simple traversal requires only n steps. The time required by a random walk to traverse a graph, i.e., the time to *cover* the graph, is an important measure of the efficiency of random walks: The *cover time* of a graph is the expected time taken by a random walk to visit every node of the graph at least once [5, 2]. The cover time is relevant to a wide range of algorithmic applications [21, 32, 23, 8], and methods of bounding the cover time of graphs have been thoroughly investigated [28, 3, 15, 13, 34, 27]. Several bounds on the cover time of particular classes of graphs have been obtained, with many positive results [15, 13, 24, 25, 16].

The contribution of this paper is proposing and partially answering the following question: Can multiple random walks search a graph faster than a single random walk? What is the cover time for a graph if we choose a node in the graph and run k random walks simultaneously from that node, where now the cover time is the expected time until each node has been visited at least once by at least one random walk?

The answer is far from obvious. Consider, for example, running k random walks simultaneously on a ring. If we start all k random walks at the same node, then the random walks have little choice but to follow each other around the ring, and it is simply a race to see which of them completes the trip first. We prove in Section 6 that on a ring the cover time for k random walks is only a factor of $\Theta(\log k)$ faster than the cover time for a single random walk. On the other hand, there are graphs for which k random walks can yield a surprising speed-up. Consider a "barbell" consisting of two cliques of size n joined by a simple path (see Figure 1 in Section 7). The cover time of such a graph is $\Theta(n^2)$ and its maximum is obtained when starting the walk from the central point of the path. In this graph, the bells on each end of the barbell act as a sink from which it is difficult for a single walk to escape, but if a logarithmic number of random walks start at the center of the barbell, each bell is likely to attract at least one random walk, which will cover that part of the graph. We prove in Section 7 that if we run $k = O(\log n)$ random walks in parallel, starting from the center, then the cover time decreases by a factor of n from $\Theta(n^2)$ to $O(n)$, which corresponds to a speed-up exponential in k.

The main result of this paper—summarized in Table 1—is that, in spite of these examples, a linear speed-up is possible for almost all interesting graphs as long as k is not too big. In Section 4, we prove that if there is a large gap between the cover time and the hitting time of a graph, where hitting time is the expected time for a random walk to move from u to v for any two nodes u and v in the graph, then k random walks cover the graph k times faster than a single random walk for k sufficiently small (see Theorems 4 and 5). Graphs that fall into this class include complete graphs, expanders, d-dimensional grids and hypercubes, d-regular balanced trees, and several types of random graphs. In the important special case of expanders, we can actually prove a linear speed-up for $k \leq n$ and not just $k \leq \log n$. While we demonstrate a relationship between the cover time and the hitting time, we also demonstrate a relationship between the cover time and the mixing time (see Theorem 9), which leads us to wonder whether there is some other property of a graph that characterizes the speed-up achieved by multiple random walks more crisply than hitting and mixing times.

There are many open problems to consider. Returning to our opening example of hunters tracking prey on a graph, for the sake of performing an analysis, our results essentially assume that the hunters all start on the same node and that the prey does not move. We believe the qualitative nature of our results continues to hold when hunters start on different nodes (a problem considered in part in [14, 11, 20]), but it is an interesting question to consider how the prey's movement might affect our results. Furthermore, our solution implicitly assumes that the hunters have no way to communicate or coordinate their movements and do not make use of any "breadcrumbs" left behind at a node by one hunter to provide feedback to other hunters visiting the same node later. In an ad-hoc wireless network, for example, allowing limited (possibly) unreliable communication among nearby hunters might change the analysis in interesting ways. Finally, one of our motivations for considering randomization in the first place was the unknown nature of the graph, but the more powerful motivation was the general desire for robust algorithms in the face of a dynamically changing graph. There are many interesting ways to formulate this problem, and actually analyzing the performance of concurrent random walks in dynamic networks would be in itself an interesting problem.

1.1 Related work

A related problem was previously studied in the context of algorithms for solving undirected s-t connectivity, the problem of deciding whether two given vertices s and t are connected in an undirected graph. The key step in many of these algorithms is to identify large subsets of connected vertices and to shrink the graph accordingly. The algorithms use short random or pseudorandom walks to identify such subsets. These walks are either starting from all the vertices of G or from a suitably chosen sample of its vertices. Deterministic algorithms concerned with the amount of used space [29, 7] use pseudorandom walks started from all the vertices of G. Parallel randomized algorithms, e.g., [26, 22], use short random walks from each vertex of G. Although there seems to be a deeper connection to our problem, these techniques do not seem to provide any results directly related to our question of interest.

However, a problem closer to ours is considered in a sequence of papers on time-space trade-offs for solving s-t-connectivity [14, 11, 20]. Algorithms in this area choose first a random set of representatives and then perform short random walks to discover connectivity between the representatives. A part of the analysis in [14] by Broder *et al.* is calculating the expected number of steps needed to cover the whole graph. Indeed, Broder *et al.* state as one of their main results that the expected number of steps taken by k random walks starting from k vertices chosen according to the stationary distribution to cover the whole graph is $O(\frac{m^2 \log^3 n}{k^2})$, where m is the number of edges and n is the number of vertices of the graph [14]. Barnes and Feige in [11, 20] consider different starting distributions that give a better time-space trade-off for the s-t-connectivity algorithm but they do not state any explicit bound on the cover time by k random walks. In contrast, in this work, we formulate our interest in comparison between the expected cover time of a single walk and of k random walks.

Although our work focuses on covering the graph starting from a single vertex, under certain conditions our results yield improved bounds on the cover time starting from the stationary distribution. In particular, for graphs with fast mixing time, Lemma 19 yields the bound $O((n \log n)/k)$ on the cover time of k random walks starting from the stationary distribution on an expander and the proof of Theorem 9 gives bound $O((nt_m \log^2 n)/k)$ on the cover time of k random walks starting from the stationary distribution on graphs with mixing time t_m. Indeed, our proofs in Section 4 do not depend on the starting distribution so similar results can be stated for k walks starting from an arbitrary probabilistic distribution.

Table 1: Results summary (for any constant $\epsilon > 0$)

Graph family name	Cover time C	Hitting time h_{max}	Mixing time t_m	Speed up S_k (order) lower bound	upper bound
cycle	$n^2/2$	$n^2/2$	$O(n^2)$	$\log(k)$	$\log(k)$
2-dimensional grid	$\Theta(n \log^2 n)$	$\Theta(n \log n)$	$\Theta(n)$	k, $k < O(\log^{1-\epsilon} n)$	
d-dimensional grid, $d > 2$	$\Theta(n \log n)$	$\Theta(n)$	$\Theta(n^{2/d})$	k, $k < O(\log^{1-\epsilon} n)$	
hypercube	$\Theta(n \log n)$	$\Theta(n)$	$\log n \log \log n$	k, $k < O(\log^{1-\epsilon} n)$	
complete graph	$\Theta(n \log n)$	$\Theta(n)$	1	k, $k < n$	k, $k < n$
expanders	$\Theta(n \log n)$	$\Theta(n)$	$\log n$	$\Omega(k)$, $k < n$	
E-R Random graph	$\Theta(n \log n)$	$\Theta(n)$	$\log n$	k, $k < O(\log^{1-\epsilon} n)$	

2. PRELIMINARIES

Let us begin with a quick review of asymptotic notation, like $o(1)$, as used in this paper: $f(n) = O(g(n))$ if there exist positive numbers c and N, such that $f(n) \leq cg(n), \forall n \geq N$. $f(n) = \Omega(g(n))$ if there exist positive numbers c and N, such that $f(n) \geq cg(n), \forall n \geq N$. $f(n) = \Theta(g(n))$ if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$. $f(n) = o(g(n))$ if $\lim_{n\to\infty} f(n)/g(n) = 0$ and $f(n) = \omega(g(n))$ if $\lim_{n\to\infty} f(n)/g(n) = \infty$.

Let $G(V, E)$ be an undirected graph, with V the set of nodes and E the set of edges. Let $n = |V|$ and $m = |E|$. For $v \in V$, let $N(v) = \{u \in V \mid (v, u) \in E\}$ be the set of neighbors of v, and let $\delta(v) = |N(v)|$ be the degree of v. A δ-regular graph is a graph in which every node has degree δ.

Let $X_i = \{X_i(t) : t \geq 0\}$ be a *simple random walk* starting from node i on the state space V with *transition matrix* Q. When the walk is at node v, the probability to move in the next step to u is $Q_{vu} = \Pr(v, u) = \frac{1}{\delta(v)}$ for $(v, u) \in E$ and 0 otherwise.

Let $\tau_i(G)$ of a graph G be the time taken by a simple random walk starting at i to visit all nodes in G. Formally $\tau_i = \min\{t : \{X_i(1), \ldots, X_i(t)\} = V\}$ and clearly this is a stopping time and therefore a random variable. Let $C_i = E[\tau_i]$ be the expected number of steps for the simple random walk starting at i to visit all the nodes in G. The *cover time* $C(G)$ of a graph G is defined formally as $C(G) = \max_i C_i$. The *cover time* of graphs and methods of bounding it have been extensively investigated [28, 3, 15, 13, 34, 5], although much less is known about the variance of the cover time. Results for the cover time of specific graphs vary from the *optimal cover time* of $\Theta(n \log n)$ associated with the complete graph K_n to the worst case of $\Theta(n^3)$ associated with the lollipop graph [19, 18].

The *hitting time*, $h(u, v)$, is the expected time for a random walk starting at u to arrive to v for the first time. Let $h_{\max}$ be the maximum $h(u, v)$ over all ordered pairs of nodes and let $h_{\min}$ to be defined similarly. The following theorem provides fundamental bound on the cover time $C(G)$ in terms of $h_{\max}$ and h_{min}.

Theorem 1 (Matthews' Theorem [28])**.** *For any graph G,*

$$h_{min} \cdot H_n \;\leq\; C(G) \;\leq\; h_{\max} \cdot H_n$$

where $H_k = \ln(k) + \Theta(1)$ is the k-th harmonic number.

Notice that this bound is not always tight, since in the line, for example, we have $C(G) = h_{\max}$.

For an integer $t > 0$, a graph G and its vertices u and v, let $p^t_{u,v}$ be the probability that a simple random walk starting from vertex u is at vertex v at time t and let $\pi(v)$ denote the probability of being at v under the stationary distribution of G. By *mixing time t_m of G*, we understand the smallest integer $t > 0$ such that for all vertices u in G, $\sum_v |p^t_{u,v} - \pi(v)| < 1/\mathrm{e}$.

2.1 k-Random Walks: Cover Time and Speed-up

Let us turn our attention to the case of k parallel independent random walks. We assume that all walks start from the same node and we are interested in the performance of such a system. The natural extension to the definition of cover time is the k cover time: Let τ^k_i be the random time taken by k simple random walks, all starting at i at $t = 0$, to visit all nodes in G (i.e., the time by which each node has been visited by at least one of the walks). Let $C^k_i = E[\tau^k_i]$ be the expected cover time for k walks starting from i. For a graph G, let $C^k(G) = \max_i C^k_i(G)$ be the k-walks' cover time. In practice, we would like to bound the speed-up in the expected cover time achieved by k walks:

Definition 2. *For a graph G and an integer $k > 1$, the* speed-up*, $S^k(G)$, on G, is the ratio between the cover time of a single random walk and the cover time of k random walks, namely, $S^k(G) = \frac{C(G)}{C^k(G)}$.*

Note that speed-up on a graph is a function of k and of the graph. When k and/or graph is understood from the context we may not mention them explicitly.

3. STATEMENT OF OUR RESULTS

We show that k random walks can cover a graph k times faster than a single random walk on a large class of graphs, a class that includes many important and practical instances. We begin with a simple statement of linear speed-up on simple graphs, but as we broaden the class of graphs considered, our statements of speed-up become more involved. We begin with a linear speed-up on cliques and expanders:

Theorem 3. *For $k \leq n$ and for a graph G that is either a complete graph on n vertices or an expander the speed-up is $S^k(G) = \Omega(k)$.*

We can show a linear speed-up on other graphs, as well, but to do so we must bound k, the number of random walks. Which bound we use depends on Matthews' bound.

When Matthews' bound is tight, we can prove a linear speed-up for k as large as $k \leq \log n$. Our proof depends on a generalization of Matthews' bound for multiple random walks: $C^k(G) \leq \frac{\mathrm{e}+o(1)}{k} \cdot h_{\max} \cdot H_n$ (see Theorem 13). Since Matthews' bound is known to be tight for the complete graph, expanders [15], d-dimensional grids for $d \geq 2$ [15], d-regular balanced trees for $d \geq 2$ [33], Erdős-Rényi random graphs [16], and random geometric graphs [9] (in the last two cases, for choice of parameters that guarantee connectivity with high probability), the following result shows that $k \leq \log n$ random walks yield a linear speed-up for a large class of interesting and useful graphs:

Theorem 4. *If $C(G) = \Theta(h_{\max} \log n)$, then $S^k(G) = \Omega(k)$ for all $k \leq \log n$.*

When Matthews' bound is not tight, we must proceed more indirectly and bound k in terms of *the gap*. Let $g(n) = \frac{C}{h_{\max}}$ be the gap between the cover time and the maximum hitting time. We find it remarkable that, using this gap, we can prove a nearly linear speed-up for k less than $g(n)$ without knowing the actual cover time.

Theorem 5. *If $g(n) = \frac{C(G)}{h_{\max}} \to \infty$ and $k \leq O(g^{1-\epsilon}(n))$ for some $\epsilon < 1$, then $C^k(G) = \frac{C(G)}{k} + o(\frac{C(G)}{k})$, and $S^k(G) \geq k - o(k)$.*

These results raise several interesting questions about speed-ups on graphs in general: is k an upper bound on the best possible speed-up, does proving a linear speed-up generally require bounding k, and what really characterizes the best possible speed-up?

For the first question, we have been unable to prove that k is an upper bound on the best possible speed-up. We do know that a wide range of speed-ups is possible, and that sometimes the speed-up can be much less than k. The following result shows that the speed-up on a cycle is limited to $\log k$.

Theorem 6. *For all $k < \mathrm{e}^{n/4}$, the speed-up on the cycle L_n with n vertices is $S^k(L_n) = \Theta(\log k)$.*

On the other hand, it is possible there are graphs for which the speed-up is much more than k. For example, the following result shows that, when the walk starts at the node in the center of the barbell graph (we cannot prove this is true from other nodes in the graph), the speed-up is *exponential* in k:

Theorem 7. *For a barbell graph B_n on n vertices (see Section 7 for a definition) if v_{c} is the center of the barbell then $C_{v_{\mathrm{c}}} = \Theta(n^2)$ but $C^k_{v_{\mathrm{c}}} = O(n)$ for $k = \Theta(\log n)$.*

For the second question, proving a linear speed-up in general does indeed require bounding k. In fact, the situation turns out to be rather complex, since the speed-up depends not only on the graph itself, but also on the relationship between the size of the graph and k. For example, using Theorem 6, we can show that there may be a full spectrum of speed-up behaviors even for a single graph:

Theorem 8. *Let G be a two dimensional $\sqrt{n} \times \sqrt{n}$ grid on the torus (for which Matthews' bound is tight).*

1. *For $k \leq \log n$, the speed-up is $S^k(G) = \Omega(k)$*
2. *For $k \geq \log^3 n$ the speed-up is $S^k(G) = o(k)$.*

Finally, what property of a graph determines the speed-up? We do not have a complete answer to this question. We are able to relate the speed-up on a graph to the ratio between cover time of the graph and the maximal hitting time of the graph as seen in Theorem 5 and further also to the mixing time of the graph. Intuitively if a graph has a fast mixing time then the random walks spread in different parts of the graph and explore it essentially independently.

Theorem 9. *Let G be a d-regular graph. If the mixing time of G is t_m then for $k \leq n$ the speed-up is $S^k = \Omega(\frac{k}{t_m \ln n})$*

To this end, questions regarding minimal and maximal bounds on the speed-up as a function of k remain open, but we do conjecture that speed-up is at most linear and at least of logarithmic order:

Conjecture 10. *For any graph G and any $k \geq 1$, $S^k(G) \leq O(k)$.*

Conjecture 11. *For any graph G and any $n \geq k \geq 1$, $S^k(G) \geq \Omega(\log k)$.*

4. LINEAR SPEED-UP

Linear speed-up in a clique follows from folklore, and we will show linear speed-up in an expander in Section 4.1; we begin by stating the simple example for later use:

Lemma 12. *For $k \leq n$ and a clique K_n of size n the speed-up is $S^k(K_n) = k$ (up-to a rounding error).*

Proof. In the lemma we restrict k to be less than n to avoid rounding problems and for simplicity we also assume self loops in the clique. We will prove this using a coupon collector argument. Let C be the number of purchases needed to collect n different coupons. Consider the case where a fair mom decides to help her k kids to collect the coupons. Each time she buys a cereal and gets a coupon she gives it to the next-in-turn son in a round-robin fashion (i.e. kid $i \bmod k$ gets the coupon from step i). Clearly, in expectation, after C visits to the grocery store mom got all the different coupons. Note that each child had his own independent coupon collecting process, and each have the same number of coupons (plus-minus one). □

We now show a linear speed-up in a much larger class of graphs, as long as $k \leq \log n$. We begin with Matthews' upper bound $C(G) \leq h_{\max} \cdot H_n$ for the cover time by a single random walk, and generalize the bound to show that k random walks improve Matthews' bound by a linear factor:

Theorem 13 (Baby Matthew Theorem)**.** *If G is a graph on n vertices and $k \leq \log n$, then*

$$C^k(G) \leq \frac{\mathrm{e} + o(1)}{k} \cdot h_{\max} \cdot H_n.$$

Proof. Let the starting vertex u of the k-walk be chosen. Fix any other vertex v in the graph G. Recall, for any two vertices u', v' in G, $h(u', v') \leq h_{\max}$. Thus by Markov inequality, Pr[a random walk of length $\mathrm{e}h_{\max}$ starting from u does not hit v] $\leq 1/\mathrm{e}$. Hence for any integer $r > 1$, the probability that a random walk of length $\mathrm{e}rh_{\max}$ does not visit v is at most $1/\mathrm{e}^r$. (We can view the walk as r independent trials to visit v.) Thus the probability that a random k-walk of length $\mathrm{e}rh_{\max}$ starting from u does not visit v is at most $1/\mathrm{e}^{kr}$. Set $r = \lceil (\ln n + 2 \ln \ln n)/k \rceil$. Then the probability that a random k-walk of length $\mathrm{e}rh_{\max}$ does not visit v is at most $1/(n \ln^2 n)$. Thus with probability at least $1 - (1/\ln^2 n)$ a random k-walk visits all vertices of G starting from u. Together with Matthews' bound $C(G) \leq h_{\max} H_n$, we can bound the k-cover time of G by $C^k(G) \leq \mathrm{e}rh_{\max}(1 + 1/\ln^2) + C(G)/\ln^2 n \leq (\mathrm{e} + o(1))h_{\max} H_n/k$. The theorem follows. □

When Matthews' bound is tight, the cover time $C(G) = \Theta(h_{\max} \log n)$, and the linear speed-up is an immediate corollary of Theorem 13:

Theorem 4. *If $C(G) = \Theta(h_{\max} \log n)$, then $S^k(G) = \Omega(k)$ for all $k \leq \log n$.*

When Matthews' bound is not tight, the proofs become more complex. We begin with the following result expressing the k-walk cover time in terms of the single-walk cover and hitting times:

Theorem 14. *For any graph G of size n large enough and for any function $f(n) \in \omega(1)$*

$$C^k(G) \;\leq\; \frac{(1+o(1))}{k} \cdot C(G) + (3\log k + 2f(n)) \cdot h_{\max}.$$

The proof is at the end of the section. In this case, we get at least an order of linear speed-up when this upper bound is dominated by the left term. Choosing $f(n)$ sufficiently small, informal calculation shows this happens when $\log k \cdot h_{\max} \leq C/k$ or $k \log k \leq C/h_{\max}$, which happens when $k = (C/h_{\max})^{1-\epsilon}$. Once again, when Matthews' bound is tight and $C/h_{\max} = \log n$ we have the following approximation to our previous result, which improves the linear speed-up constant from $1/e$ to 1 at the cost of a slight reduction in the choice of applicable k:

Corollary 15. *If $C = \Theta(h_{\max} \log n)$ and $k = O(\log^{1-\epsilon} n)$ for some $\epsilon < 1$, then $C^k = \frac{C}{k} + o(\frac{C}{k})$, and $S^k(G) \geq k - o(k)$.*

When Matthews' bound is not tight, we have the following result expressed directly in terms of the gap $g(n) = \frac{C}{h_{\max}}$ between the cover time and the hitting time:

Theorem 5. *If $g(n) = \frac{C(G)}{h_{\max}} \to \infty$ and $k = O(g^{1-\epsilon}(n))$ for some $\epsilon < 1$, then $C^k(G) = \frac{C}{k} + o(\frac{C}{k})$, and $S^k(G) \geq k - o(k)$.*

Proof. Set $f(n) \in \omega(1)$ in Theorem 14 to be $\log(g(n))$, and the claim follows. □

We now prove Theorem 14. Our main technical tool conceptually different from our previous proofs is the following lemma.

Lemma 16. *Let G be a graph and $u_1, \ldots, u_k$ be some of its vertices, not necessarily distinct. Let T_c and p_c be such that a random walk of length T_c starting from u_1 visits all vertices of G with probability at least p_c. Let T_h and p_h be such that for any two vertices u and v of G, a random walk of length T_h starting from u visits v with probability at least p_h. Let $\ell > 1$ be an integer. Then a random k-walk of length $T_c/k + \ell T_h$ starting from vertices $u_1, \ldots, u_k$ covers G with probability at least $p_c(1 - k(1-p_h)^\ell)$.*

Proof. The proof is conceptually simple. We introduce here a little bit of notation to describe it formally. For a sequence of vertices $\vec{c} = (c_0, c_1, \ldots, c_t)$ and a random walk X on G starting from c_0, $\vec{c} \sqsubseteq X$ denotes the event $\bigwedge_{i=0}^{t} X(i) = c_i$. For two sequences $\vec{c} = (c_0, \ldots, c_t)$ and $\vec{d} = (d_0, \ldots, d_{t'})$, where $c_t = d_0$ we denote by $\vec{c} \circ \vec{d} = (c_0, \ldots, c_t, d_1, \ldots, d_{t'})$. It is straightforward to verify, if X is a random walk starting from c_0 and Y is an independent random walk starting from d_0, then $\Pr[\vec{c} \sqsubseteq X \;\&\; \vec{d} \sqsubseteq Y] = \Pr[\vec{c} \circ \vec{d} \sqsubseteq X]$. Last, for an integer $m \geq 1$ and a sequence $\vec{c} = (c_0, c_1, \ldots, c_{km-1})$, $\vec{c}_{k,i}$ denotes the subsequence $(c_{(i-1)m}, \ldots, c_{im-1})$ for $0 \leq i \leq k$.

WLOG T_c is divisible by k. Clearly, the probability that a random k-walk $(X_1, \ldots, X_k)$ of length $T_c/k + \ell T_h$ on G starting from vertices $u_1, \ldots, u_k$ covers all of G can be lower-bounded by

$$p = \Pr\left[\bigvee_{\vec{c}, \vec{h}_2, \ldots, \vec{h}_k} \begin{array}{r} \vec{c}_{k,1} \sqsubseteq X_1 \;\&\; \vec{h}_2 \circ \vec{c}_{k,2} \sqsubseteq X_2 \;\&\; \cdots \\ \vec{h}_k \circ \vec{c}_{k,k} \sqsubseteq X_k \end{array}\right],$$

where $\vec{c}$ is taken from the set of all sequences of vertices from G corresponding to walks of length T_c on G that start in u_1 and cover whole G, and $\vec{h}_i$ is taken from the set of all sequences of vertices from G corresponding to walks of length at most ℓT_h that start in u_i and hit $c_{(i-1)T_c/k}$ for the first time only at their end. It is easy to verify that all the events in the union are disjoint. Hence,

$$\begin{aligned} p &= \sum_{\vec{c}, \vec{h}_2, \ldots, \vec{h}_k} \Pr\left[\begin{array}{r} \vec{c}_{k,1} \sqsubseteq X_1 \;\&\; \vec{h}_2 \circ \vec{c}_{k,2} \sqsubseteq X_2 \;\&\; \cdots \\ \vec{h}_k \circ \vec{c}_{k,k} \sqsubseteq X_k \end{array}\right] \\ &= \sum_{\vec{c}, \vec{h}_2, \ldots, \vec{h}_k} \Pr\left[\vec{c} \sqsubseteq X_1 \;\&\; \vec{h}_2 \sqsubseteq X_2 \;\&\; \cdots \vec{h}_k \sqsubseteq X_k\right] \\ &= \sum_{\vec{c}, \vec{h}_2, \ldots, \vec{h}_k} \Pr[\vec{c} \sqsubseteq X_1] \cdot \Pr[\vec{h}_2 \sqsubseteq X_2] \cdots \Pr[\vec{h}_k \sqsubseteq X_k] \\ &= \sum_{\vec{c}} \Pr[\vec{c} \sqsubseteq X_1] \cdot \sum_{\vec{h}_2} \Pr[\vec{h}_2 \sqsubseteq X_2] \cdots \sum_{\vec{h}_k} \Pr[\vec{h}_k \sqsubseteq X_k], \end{aligned}$$

where the third equality follows from the independence of the walks. By our assumption $\sum_{\vec{c}} \Pr[\vec{c} \sqsubseteq X_1] \geq p_c$. Since $(1-a)(1-b) \geq (1-a-b)$ for $a, b \leq 1$, to conclude the lemma it suffices to argue that $\sum_{\vec{h}_i} \Pr[\vec{h}_i \sqsubseteq X_i] \geq 1 - (1-p_h)^\ell$ for all i. Notice that $\sum_{\vec{h}_i} \Pr[\vec{h}_i \sqsubseteq X_i] = \Pr[$ a random walk of length ℓT_h starting from u_i visits $c_{(i-1)T_c/k}]$. Since a random walk of length T_h fails to visit $c_{(i-1)T_c/k}$ with probability at most $1 - p_h$ regardless of its starting vertex, a random walk of length ℓT_h fails to visit $c_{(i-1)T_c/k}$ with probability at most $(1-p_h)^\ell$. The lemma follows. □

Next, we use the following bound on the concentration of the cover time by Aldous [4]:

Theorem 17 ([4])**.** *For the simple random walk on G, starting at i, if $C_i/h_{\max} \to \infty$ then $\tau_i/C_i \xrightarrow{p} 1$.*

Equipped with the proper tools we are ready to prove Theorem 14.

Proof of Theorem 14. If the conditions of Theorem 17 do not hold then the cover time and hitting time are on the same order and Theorem 14 gives a trivial (not tight) upper bound. Assume the conditions of Theorem 17 holds. Theorem 17 implies that $\Pr[\tau_u/C_u > 1 + \delta_n] \leq \epsilon_n$ where $\delta_n, \epsilon_n \to 0$ as the size of the graph goes to infinity. Thus $\Pr[$a random walk of length $(1+o(1)) \cdot C$ covers $G] \geq 1 - o(1)$. By Markov bound, for a fixed vertex v of the graph, $\Pr[$a random walk of length $2h_{\max}$ visits vertex $v] \geq 1/2$. If we set $\ell = \log k + \omega(1)$, then Lemma 16 implies that a random k-walk of length $L = \frac{(1+o(1))C}{k} + (\log k + \omega(1))2h_{\max}$ covers G with probability at least $(1-o(1)) \cdot (1 - k2^{-\ell}) = (1-o(1)) \cdot \left(1 - \frac{1}{\omega(1)}\right) = 1 - o(1)$. Here each of the k random walks may start at a different vertex. Thus a walk of length $i \cdot L$ does not cover G with probability at most $[o(1)]^i$ so the cover time of G can be bounded by $L\sum_i i\cdot[o(1)]^i = L\cdot\frac{1}{1-o(1)} = L\cdot(1+o(1))$. □

4.1 Linear speed-up on expanders

In this section we prove that for the important special case of expanders, there is a linear speed-up for k as large as $k\le n$:

Theorem 18. *If G is an expander, then the speed-up $S^k(G)=\Omega(k)$ for $k\le n$.*

An (n,d,λ)-graph is a d-regular graph G on n vertices so that the absolute value of every nontrivial eigenvalue of the adjacency matrix of G is at most λ. It is well known (see [6]) that a d-regular graph on n vertices (with a loop in every vertex) is an expander (that is, any set X of at most half the vertices has at least $c|X|$ neighbors outside the set, where $c>0$ is bounded away from zero), if and only if there is a fixed λ bounded away from d so that G is an (n,d,λ)-graph. Since the rate of convergence of a random walk to a uniform distribution is determined by the spectral properties of the graph it will be convenient to use this equivalence and prove that random walks on (n,d,λ)-graphs, where λ is bounded away from d, achieve linear speed up. In what follows we make no attempt to optimize the absolute constants, and omit all floor and ceiling signs whenever these are not crucial.

Lemma 19. *Let G be an (n,d,λ)-graph. Put $s=\frac{\ln(2n)}{\ln(d/\lambda)}$ and $b=\frac{\lambda}{d-\lambda}$. Then, for every two vertices u,v of G, the probability that a random walk of length $2s$ starting at u, covers v is at least $\frac{s}{2n+4s+4bn}$.*

Proof. For each i, $s<i\le 2s$, let Y_i be the indicator random variable whose value is 1 iff the walk starting at u visits v at step number i. Let $Y=\sum_{i=s+1}^{2s}Y_i$ be the number of times the walk visits v during its last s steps. Our objective is to show that the probability that Y is positive is at least $\frac{s}{2n+4s+4bn}$. To do so, we estimate the expectation of Y and of Y^2 and use the fact that by Cauchy-Schwartz

$$\begin{aligned}\Pr[Y>0]=\sum_{j>0}\Pr[Y=j]&\ge\frac{(\sum_{j>0}j\Pr[Y=j])^2}{\sum_{j>0}j^2\Pr[Y=j]}\\&=\frac{(E(Y))^2}{E(Y^2)}\end{aligned}\tag{1}$$

By linearity of expectation $E(Y)=\sum_{i=s+1}^{2s}E(Y_i)$. The expectation of Y_i is the probability the walk visits v at step i. This is precisely the value of the coordinate corresponding to v in the vector A^iz, where A is the stochastic matrix of the random walk, that is the adjacency matrix of G divided by d, and z is the vector with 1 in the coordinate u and 0 in every other coordinate. Writing z as a sum of the constant $1/n$-vector z_1 and a vector z_2 whose sum of coordinates is 0, and using the fact that $Az_1=z_1$ and that the ℓ_2-norm of A^iz_2 satisfies $||A^iz_2||\le(\frac{\lambda}{d})^i$ we conclude, by the definition of s, that each coordinate of A^iz deviates from $1/n$ by at most $\frac{1}{2n}$.

It thus follows that

$$E(Y)\ge\frac{s}{2n}.\tag{2}$$

By linearity of expectation

$$E(Y^2)=\sum_{i=s+1}^{2s}E(Y_i)+2\sum_{s<i<j\le 2s}E(Y_iY_j)$$

Note that $E(Y_iY_j)$ is precisely the probability that the walk visits v at step i and at step j. This is the probability that it visits v at step i, times the conditional probability that it visits v at step j given that it visits it at step i. This conditional probability can be estimated as before, showing that it deviates from $1/n$ by at most $(\lambda/d)^{j-i}$. It thus follows that

$$\begin{aligned}E(Y^2)&\le E(Y)+2\sum_{i=s+1}^{2s}E(Y_i)(\frac{s}{n}+\sum_{r>0}(\lambda/d)^r)\\&\le E(Y)[1+\frac{2s}{n}+2\frac{\lambda}{d-\lambda}].\end{aligned}\tag{3}$$

Plugging the estimates (2) and (3) in (1) we conclude that

$$\begin{aligned}\Pr[Y>0]&\ge\frac{(E(Y))^2}{E(Y)[1+2s/n+2\lambda/(d-\lambda)]}\\&\ge\frac{s/(2n)}{1+2s/n+2b}=\frac{s}{2n+4s+4bn}.\end{aligned}$$

This completes the proof. □

Corollary 20. *Let G be an (n,d,λ)-graph and define $s=\frac{\ln(2n)}{\ln(d/\lambda)}$, $b=\frac{\lambda}{d-\lambda}$. Suppose $n\ge 2s$, and let k be an integer so that $\frac{16(b+1)n\ln n}{k}>2s$. For any two fixed vertices u and v of G, the probability that v is not covered by at least one of k independent random walks starting at u, each of length $t=\frac{16(b+1)n\ln n}{k}$, is smaller than $\frac{1}{n^2}$.*

Proof. Break each of the walks into $\frac{t}{2s}$ sub-walks, each of length $2s$. By Lemma 19, for each of these sub-walks, the probability it covers v is at least $\frac{s}{2n+4s+4bn}\ge\frac{s}{4(b+1)n}$. Note that this estimate holds for each specific sub-walk, even after we expose all previous sub-walks, as given this information it is still a random walk of length $2s$ starting at some vertex of G, and this initial vertex is known once the previous sub-walks are exposed. It follows that the probability that v is not covered is at most

$$(1-\frac{s}{4(b+1)n})^{kt/2s}<\mathrm{e}^{-kt/(8(b+1)n}=\mathrm{e}^{-2\ln n}=\frac{1}{n^2},$$

as needed. □

In the notation of the above corollary, the k random walks of length t starting at u cover the whole expander with probability at least $1-1/n$. Since the usual cover time of the expander is $O(n\ln n)$ it follows that the expected length of the walks until they cover the graph does not exceed $t+\frac{1}{n}O(n\ln n)\le O(t)$.

Note that for every fixed b, the total length of all k walks in the last corollary is $O(n\ln n)$, and that the assumption $\frac{16(b+1)n\ln n}{k}>2s=2\frac{\ln(2n)}{\ln(d/\lambda)}$ holds for every k which does not exceed $b'n$ for some absolute constant b' depending only on b (as $d/\lambda=1+1/b$). This shows that k random walks on n-vertex expanders achieve speed-up $\Omega(k)$ for all $k\le n$.

5. SPEED-UP AND MIXING TIME

Random walks on expanders converge rapidly to the stationary distribution. For graphs with fast mixing times, like expanders, the following theorem gives a second bound on the speed-up in terms of mixing time.

Theorem 9. *Let G be a d-regular graph. If the mixing time of G is t_m then for $k \le n$ the speed-up is $S^k = \Omega(\frac{k}{t_m \ln n})$*

Proof. Let G be a d-regular graph of size n. We show that the expected cover time of G by a random k-walk is $O(\frac{t_m n \ln^2 n}{k})$. As a cover time of any graph is at least $n \ln n$ the theorem follows.

In this proof we represent a random k-walk on G by an infinite sequence of random variables $X_0, X_1, \dots$, where X_i is the position of the $1 + (i \bmod k)$-th token at step $\lfloor i/k \rfloor + 1$. Define the random variables $Y_i = X_{\lfloor i/k \rfloor k \cdot 6t_m \ln n + (i \bmod k)}$. Hence, Y_i's correspond to the position of the k-walk after every $6t_m \ln n$ steps. Let a random variable Y'_i be Y_i conditioned on a specific outcome of $Y_0, \dots, Y_{i-k}$. Since t_m is the mixing time of G and the stationary distribution of a random walk on G is uniform (G is d-regular), the statistical distance of Y'_i from the uniform distribution on G is at most $(1/\mathrm{e})^{6 \ln n} \le 1/n^6$. In particular, for any vertex v of G, $|\Pr[Y'_i = v] - 1/n| \le 1/n^6$.

Thus, for any $1 < \ell \le n^3$ and any sequence $v_1, \dots, v_\ell$ of vertices

$$(1/n - /n^6)^\ell \le \Pr[Y'_1 Y'_2 \cdots Y'_\ell = v_1 \cdots v_\ell] \le (1/n + 1/n^6)^\ell$$

Hence,

$$1/n^\ell \cdot (1 - 1/n^2) \le \Pr[Y'_1 Y'_2 \cdots Y'_\ell = v_1 \cdots v_\ell] \le 1/n^\ell \cdot (1 + 2/n^2).$$

One can easily show (see the proof of Theorem 26) that the probability that a clique of size n is not covered within $10n \ln n$ steps by a random 1-walk is at most $1/n^9$. By the above bound distribution of $Y'_1 Y'_2 \cdots Y'_\ell$, for $1 < \ell \le n^3$ is close to a distribution of a random walk on a clique. Hence, unless $Y'_1, Y'_2, \dots, Y'_{10n \ln n}$ does not hit all the vertices of G, we can bound the expected cover time of G by $(6t_m \ln n) \cdot C^k(K_n) \cdot (1 + 2/n^2)$. If $Y'_1, Y'_2, \dots, Y'_{10n \ln n}$ does not hit all the vertices of G we can bound the cover time of G by the trivial bound $O(n^3)$. Since $C^k(K_n) = O(n \ln n/k)$ the claim follows. $\square$

6. LOGARITHMIC SPEED-UP

So far we have seen only cases where the speed-up in cover time achieved by multiple random walks is considerable, i.e., at least linear. In this section we show that this is not always the case and that the speed-up may be as low as logarithmic in k. The cover time of a cycle L_n on n vertices is $\Theta(n^2)$. We prove the following claim.

Theorem 6. *For any integer n and $k < \mathrm{e}^{n/4}$, the speed-up on the cycle with n vertices is $S^k(L_n) = \Theta(\log k)$.*

Hence for a cycle even a moderate speed-up of $\omega(\log n)$ requires super-polynomially many walks, and to achieve speed-up of n^ϵ one requires $2^{\Omega(n^\epsilon)}$ walks. The theorem follows from the following two lemmas.

Lemma 21. *Let $s > 1$ and $k \ge 1$ be such that $C^k \le n^2/s$ for a cycle of length n. Then $k \ge \mathrm{e}^{s/16}/8$.*

Proof. Assume that $C^k \le n^2/s$ and we will prove that $k \ge \mathrm{e}^{s/16}/8$. Pick an arbitrary vertex v of the graph. Clearly, the cover time starting from the vertex v is $C^k_v \le n^2/s$. Let a random variable T_v be the cover time of a random k-walk starting from v. By Markov inequality, $\Pr[T_v \ge 2n^2/s] \le 1/2$. Hence, with probability at least $1/2$ one of the k walks reaches the vertex $v_{n/2}$ that is at distance $n/2$ from v in at most $2n^2/s$ steps. For a single walk, if it reaches $v_{n/2}$ starting from v in time at most $2n^2/s$, then there is $1 \le t \le 2n^2/s$ so that the number of its steps to the right until time t differs from the number of its steps to the left by at least $n/2$. Given that this happens, with probability $1/2$ the number of steps to the right will differ from the number of steps to the left by at least $n/2$ also at time $2n^2/s$. This is because after time t we will increase the difference with the same probability as that we will decrease it since the probability of going to the left is the same as the probability of going to the right. By Chernoff bound, Pr[the number of steps to the left and to the right of a walk differs by at least $n/2$ at time $2n^2/s$] $\le 2\mathrm{e}^{-\frac{s \cdot n^2}{16n^2}} \le 2\mathrm{e}^{-s/16}$. Hence, the probability that a particular walk reaches the vertex $v_{n/2}$ during $2n^2/s$ steps is at most $4\mathrm{e}^{-s/16}$.

Thus, Pr[there exists a walk that reaches $v_{n/2}$ in time at most $2n^2/s$] $\le 4k \cdot \mathrm{e}^{-s/16}$. Since this probability must be at least $1/2$ we conclude that $\frac{\mathrm{e}^{s/16}}{8} \le k$. $\square$

Lemma 22. *Let k be large enough and n be an integer. If $k \le \mathrm{e}^{n/4}$ then $C^k \le 2n^2/\ln k$ for a cycle of length n.*

To prove this lemma we need the following folklore statement (see the appendix for the proof).

Proposition 23. *Let $c \ge 2$ be a constant. For every even integer $n \ge 16c^2$,*

$$\mathrm{e}^{-3c^2-4} \le \Pr[(c-1)\sqrt{n} \le X - n/2 \le c\sqrt{n}] \le \mathrm{e}^{-2(c-1)^2},$$

where X is a sum of n independent 0-1 random variables that are 1 with probability 1/2.

Proof of Lemma 22. To prove that $C^k \le \frac{2n^2}{\ln k}$, let $c = \sqrt{\ln k}/2$ and $\ell = n^2/4(c-1)^2$. If a single walk during a random k-walk of length ℓ on a cycle of length n makes in total at least $\ell/2 + n/2$ steps to the right then it traversed around the whole cycle. Note, $n/2 = \sqrt{\ell}(c-1)$. By the previous proposition, Pr[a single walk makes at least $\ell/2 + n/2$ steps to the right during a random walk of length ℓ] $\ge \mathrm{e}^{-3c^2-4} \ge 1/k$, for k large enough. Hence, k walks walking in parallel at random for ℓ steps fail to cover the whole cycle of length n with probability at most $(1-1/k)^k < 1/\mathrm{e}$. Thus $C^k \le \sum_{i=0}^{\infty} \frac{1}{\mathrm{e}^i}\ell = \mathrm{e}\ell/(\mathrm{e}-1) \le 2n^2/\ln k$, for k large enough. $\square$

Lemma 21 also implies the following claim.

Theorem 24. *Let $G_{n,d}$ be a d-dimensional grid (torus) on $n^{1/d} \times n^{1/d} \times \cdots n^{1/d}$ vertices, $d \ge 2$. For any k, $C^k(G_{n,d}) \ge \Omega(n^{2/d}/\log k)$.*

Proof. We prove the claim for $d = 2$. The other cases are analogous. Consider the random k-walk on a $\sqrt{n} \times \sqrt{n}$ grid (torus). We can project the position of each of the k walks to the x axis. This will give a distribution identical to a k-walk on a cycle of size $\sqrt{n}$ where in each step we make a

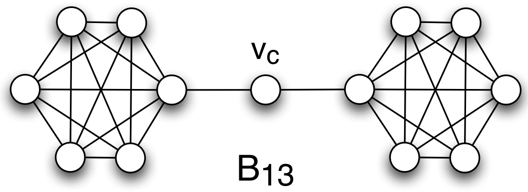

Figure 1: Example barbell graph B_{13}, v_c is the center of the barbell

step to the left with probability 1/4, step to the right with probability 1/4 and with the remaining probability 1/2 we stay at the current vertex. In order for a k-walk to cover the whole grid, this projected walk must cover the whole cycle. Thus the expected cover time of the grid must be lower-bounded by the expected cover time for a cycle of size $\sqrt{n}$ which is $\Omega(n/\log k)$ by Lemma 21. (Note the steps in which we stay at the same vertex can only increase the cover time.) □

Corollary 25. *For a 2-dimensional grid $G_{n,2}$, $S^k(G_{n,2}) \leq O(\log^2 n \log k)$.*

This corollary together with Theorem 4 implies Theorem 8.

7. EXPONENTIAL SPEED-UP

On some graphs the speed-up can be exponential in k for at least some choice of the starting point. For an odd integer $n > 1$, we define a barbell graph B_n to be a graph consisting of two cliques of size $(n-1)/2$ connected by a path of length 2 (see Figure 1). The vertex on that paths is called the *center* of B_n and the cliques are called *bells*. The expected time to cover B_n by a random walk is $\Theta(n^2)$ since once the token is in one of the cliques it takes on average $\Theta(n^2)$ steps to exit that clique. It can be shown that the maximum cover time is attained by starting the random walk from the center of B_n. We show the following theorem.

Theorem 26. *Let $n > 1$ be an odd integer, v_c be the center of B_n and $k = 20 \ln n$. The expected cover time starting from v_c satisfies $C^k_{v_c} = O(n)$.*

Hence, the speed-up in a cover time starting from a particular vertex of a k-random walk compared to a random walk by a single token may be substantially larger than k. In the case of B_n the speed-up is $\Omega(n)$ for $O(\log n)$-walks for walks starting at a particular vertex.

Proof. With high probability none of the following three events happens:

$\mathcal{E}1$ In one of the bells there are less than $4 \ln n$ tokens after the first step.

$\mathcal{E}2$ During the first $10n$ steps of the random k-walk at least $2 \ln n$ vertices return to the center.

$\mathcal{E}3$ One of the bells is not covered within the first $10n$ steps.

If none of the above events happens then each of the bells is explored by at least $2 \ln n$ tokens. Two disjoint cliques of size $m = (n-1)/2$ are each covered by a random $2 \ln n$-walk in expected time $2C^{2 \ln n}(K_m)$, by Lemma 12. So if C is the expected cover time of B_n by a random 1-walk then:

$$C^k_{v_c} \leq 2C^{2 \ln n}(K_m) + \Pr[(1)]C + \Pr[\mathcal{E}2 \cup \mathcal{E}3](10n + C).$$

We need to estimate the probabilities of the above events. By Chernoff bound,

$$\Pr[\mathcal{E}1] \leq 2\mathrm{e}^{-(16 \ln n)^2/2 \cdot 20 \ln n} < 1/n^5$$

for n large enough. A single token returns to the center of B_n within $10n$ steps with probability at most $\frac{1}{n} + \frac{10n}{m(m+1)} < \frac{22}{m}$. The probability that at least $2 \ln n$ vertices return to the center is then $< 2^{20 \ln n} \cdot (22/m)^{2 \ln n} < 1/n^5$, for n large enough. Finally, the probability that a random $2 \ln n$-walk does not cover a clique of size m in $10n$ steps is at most $m(1 - \frac{1}{m})^{20n \ln n} \leq m\mathrm{e}^{-10 \ln n} < 1/n^5$. Now since $C = O(n^2)$ and $C^{2 \ln n}(K_m) = O(n)$, we get $C^k_{v_c} = O(n)$. □

8. CONCLUSIONS AND OPEN PROBLEMS

In this paper, we have shown that many random walks can be faster than one, sometimes much faster. Our main result is that a linear speed-up is possible on a large class of interesting graphs—including complete graphs, expanders, grids, hypercubes, balanced trees, and random graphs—in the sense that $k \leq \log n$ random walks can cover an n-node graph k times faster than a single random walk. In the case of expanders, we obtain a linear speed-up even when k is as large as n. Our technique is to relate the expected cover time for k random walks to the expected cover and hitting times for a single random walk; and to observe that if there is a large gap between the single-walk cover and hitting times, then a linear speed-up is possible using multiple random walks. Using a different technique, we were able to bound the k-walk cover time in terms of the mixing time as well.

Open problems abound, despite of the progress reported here. There are the standard questions concerning improving bounds. Is it possible that the speed-up is always at most k? Our single counter example was that multiple random walks starting at the center of the barbell achieved an exponential speed-up, but perhaps the speed-up is limited to k if we start at other nodes. Is it possible that the speed-up is always at least $\log k$? We have shown that the speed-up is $\log k$ on the ring, and we conjecture this is possible on any graph.

Another source of open problems is to consider more general classes of graphs. Said in another way, our approach has been to relate the k-walk cover time to the single-walk hitting time and mixing time, but is there another property of a graph that more crisply characterizes the speed-up achieved by multiple random walks?

9. REFERENCES

[1] Alanyali, M., Saligrama, V., and Sava, O. A random-walk model for distributed computation in energy-limited network. In *In Proc. of 1st Workshop on Information Theory and its Application* (San Diego, 2006).

[2] Aldous, D. J. On the time taken by random on finite groups to visit every state. *Z. Wahrsch. Verw. Gebiete 62*, 3 (1983), 361–374.

[3] Aldous, D. J. Lower bounds for covering times for reversible markov chains and random walks on graphs. *J. Theoret. Probab. 2*, 1 (1989), 91–100.

[4] Aldous, D. J. Threshold limits for cover times. *Journal of Theoretical Probability V4*, 1 (1991), 197–211.

[5] Aleliunas, R., Karp, R. M., Lipton, R. J., Lovász, L., and Rackoff, C. Random walks, universal traversal sequences, and the complexity of maze problems. In *20th Annual Symposium on Foundations of Computer Science (San Juan, Puerto Rico, 1979)*. IEEE, New York, 1979, pp. 218–223.

[6] Alon, N. Eigenvalues and expanders. *Combinatorica 6*, 2 (1986), 83–96.

[7] Armoni, R., Ta-Shma, A., Wigderson, A., and Zhou, S. A $(\log n)^{4/3}$ space algorithm for (s, t) connectivity in undirected graphs. *Journal of the ACM 47*, 2 (2000), 294–311.

[8] Avin, C., and Brito, C. Efficient and robust query processing in dynamic environments using random walk techniques. In *Proc. of the third international symposium on Information processing in sensor networks* (2004), pp. 277–286.

[9] Avin, C., and Ercal, G. On the cover time of random geometric graphs. In *Proc. Automata, Languages and Programming, 32nd International Colloquium, ICALP05* (2005), pp. 677–689.

[10] Bar-Yossef, Z., Friedman, R., and Kliot, G. Rawms -: random walk based lightweight membership service for wireless ad hoc network. In *MobiHoc '06: Proceedings of the seventh ACM international symposium on Mobile ad hoc networking and computing* (New York, NY, USA, 2006), ACM Press, pp. 238–249.

[11] Barnes, G., and Feige, U. A spectrum of time-space tradeoffs for undirected s-t connectivity. *Journal of Computer and System Sciences 2*, 54 (1997), 305–316.

[12] Braginsky, D., and Estrin, D. Rumor routing algorthim for sensor networks. In *Proc. of the 1st ACM Int. workshop on Wireless sensor networks and applications* (2002), ACM Press, pp. 22–31.

[13] Broder, A., and Karlin, A. Bounds on the cover time. *J. Theoret. Probab. 2* (1989), 101–120.

[14] Broder, A., Karlin, A., Raghavan, P., and Upfal, E. Trading space for time in undirected s-t connectivity. In *Proc. ACM Symp. Theory of Computing* (1989), pp. 543–549.

[15] Chandra, A. K., Raghavan, P., Ruzzo, W. L., and Smolensky, R. The electrical resistance of a graph captures its commute and cover times. In *Proc. of the twenty-first annual ACM symposium on Theory of computing* (1989), ACM Press, pp. 574–586.

[16] Cooper, C., and Frieze, A. The cover time of sparse random graphs. In *Proceedings of the fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA-03)* (Baltimore, Maryland, USA, 2003), ACM Press, pp. 140–147.

[17] Dolev, S., Schiller, E., and Welch, J. Random walk for self-stabilizing group communication in ad-hoc networks. In *Proceedings of the 21st IEEE Symposium on Reliable Distributed Systems (SRDS'02)* (2002), IEEE Computer Society, p. 70.

[18] Feige, U. A tight lower bound on the cover time for random walks on graphs. *Random Structures and Algorithms 6*, 4 (1995), 433–438.

[19] Feige, U. A tight upper bound on the cover time for random walks on graphs. *Random Structures and Algorithms 6*, 1 (1995), 51–54.

[20] Feige, U. Short random walks on graphs. *SIAM Journal on Discrete Mathematics 1*, 9 (1996), 19–28.

[21] Gkantsidis, C., Mihail, M., and Saberi, A. Random walks in peer-to-peer networks. In *in Proc. 23 Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM). to appear* (2004).

[22] Halperin, S., and Zwick, U. An optimal randomized logarithmic time connectivity algorithm for the EREW PRAM. *Journal of Computer and System Sciences 53* (1996), 395–416.

[23] Jerrum, M., and Sinclair, A. The markov chain monte carlo method: An approach to approximate counting and integration. In *Approximations for NP-hard Problems, Dorit Hochbaum ed.* PWS Publishing, Boston, MA, 1997, pp. 482–520.

[24] Jonasson, J. On the cover time for random walks on random graphs. *Comb. Probab. Comput. 7*, 3 (1998), 265–279.

[25] Jonasson, J., and Schramm, O. On the cover time of planar graphs. *Electronic Communications in Probability 5* (2000), 85–90.

[26] Karger, D. R., Nisan, N., and Parnas, M. Fast connected components algorithms for the EREW PRAM. *SIAM J. Comput. 28*, 3 (1999), 1021–1034.

[27] Lovász, L. Random walks on graphs: A survey. In *Combinatorics, Paul Erdős is eighty, Vol. 2 (Keszthely, 1993)*, vol. 2 of *Bolyai Soc. Math. Stud.* János Bolyai Math. Soc., Budapest, 1996, pp. 353–397.

[28] Matthews, P. Covering problems for brownian motion on spheres. *Ann. Probab. 16*, 1 (1988), 189–199.

[29] Nisan, N., Szemerédi, E., and Wigderson, A. Undirected connectivity in $O(\log^{1.5} n)$ space. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science* (1992), pp. 24–29.

[30] Sadagopan, N., Krishnamachari, B., and Helmy, A. Active query forwarding in sensor networks (acquire). *Journal of Ad Hoc Networks 3*, 1 (January 2005), 91–113.

[31] Servetto, S. D., and Barrenechea, G. Constrained random walks on random graphs: Routing algorithms for large scale wireless sensor networks. In *Proc. of the first ACM Int. workshop on Wireless sensor networks and applications* (2002), ACM Press, pp. 12–21.

[32] Wagner, I. A., Lindenbaum, M., and Bruckstein, A. M. Robotic exploration, brownian motion and electrical resistance. *Lecture Notes in Computer Science 1518* (1998), 116–130.

[33] Zuckerman, D. Covering times of random walks on bounded degree trees and other graphs. *Journal of Theoretical Probability V2*, 1 (1989), 147–157.

[34] Zuckerman, D. A technique for lower bounding the cover time. In *Proc. of the twenty-second annual ACM*

symposium on Theory of computing (1990), ACM Press, pp. 254–259.

APPENDIX

A. BOUNDING LARGE DEVIATION

Proposition 23. *Let $c \geq 2$ be a constant. For every even integer $n \geq 16c^2$,*

$$\mathrm{e}^{-3c^2-4} \leq \Pr[(c-1)\sqrt{n} \leq X - n/2 \leq c\sqrt{n}] \leq \mathrm{e}^{-2(c-1)^2},$$

where X is a sum of n independent 0-1 *random variables that are* 1 *with probability* 1/2.

Proof. The upper bound follows from Chernoff bound. The lower bound can be derived as follows. $\Pr[(c-1)\sqrt{n} \leq X - n/2 \leq c\sqrt{n}] = \sum_{k=(c-1)\sqrt{n}}^{c\sqrt{n}} \Pr[X - n/2 = k]$. For any k, $\Pr[X - n/2 = k] = \binom{n}{n/2+k}/2^n$. We will compare $\binom{n}{n/2+k}$ with the central binomial coefficient $\binom{n}{n/2}$.

$$\begin{aligned}
\frac{\binom{n}{n/2}}{\binom{n}{n/2+c\sqrt{n}}} &= \Pi_{j=1}^{n/2}\frac{(n-j+1)}{j} \cdot \Pi_{j=1}^{n/2+c\sqrt{n}}\frac{j}{(n-j+1)} \\
&= \Pi_{j=n/2+1}^{n/2+c\sqrt{n}}\frac{j}{(n-j+1)} \\
&= \Pi_{j=1}^{c\sqrt{n}}\frac{1+\frac{2}{n}j}{(1-\frac{2}{n}(j+1))}.
\end{aligned}$$

We upper-bound this ratio as follows:

$$\begin{aligned}
\Pi_{j=1}^{c\sqrt{n}}(1+\frac{2}{n}j) &\leq \mathrm{e}^{\frac{2}{n}\sum_{j=1}^{c\sqrt{n}} j} \\
&= \mathrm{e}^{\frac{2}{n}\cdot\frac{c\sqrt{n}(c\sqrt{n}+1)}{2}} \\
&\leq \mathrm{e}^{c^2+1}.
\end{aligned}$$

Now, for $0 \leq x \leq 1/2$, $\mathrm{e}^{-2x} \leq 1 - x$. Hence,

$$\begin{aligned}
\Pi_{j=1}^{c\sqrt{n}}(1-\frac{2}{n}(j+1)) &\geq \mathrm{e}^{-\frac{4}{n}\sum_{j=1}^{c\sqrt{n}}(j+1)} \\
&\geq \mathrm{e}^{-\frac{4}{n}\cdot\frac{(c\sqrt{n}+1)(c\sqrt{n}+2)}{2}} \\
&\geq \mathrm{e}^{-2c^2-2}.
\end{aligned}$$

Thus

$$\frac{\binom{n}{n/2}}{\binom{n}{n/2+c\sqrt{n}}} \leq \mathrm{e}^{3c^2+3}.$$

Using estimates on Stirling's formula $\binom{n}{n/2} \geq \sqrt{\frac{2}{\mathrm{e}\pi n}} \cdot 2^n$, we conclude that

$$\sum_{k=(c-1)\sqrt{n}}^{c\sqrt{n}} \binom{n}{n/2+k} \geq \binom{n}{n/2}\sqrt{n}\mathrm{e}^{-3c^2-3} \geq \mathrm{e}^{-3c^2-4} \cdot 2^n.$$

The lemma follows. □