

Many Random Walks Are Faster Than One[†]

NOGA ALON^{1§}, CHEN AVIN², MICHAL KOUCKÝ^{3¶},
GADY KOZMA⁴, ZVI LOTKER² and MARK R. TUTTLE⁵

¹Sackler School of Mathematics and Blavatnik School of Computer Science,
Tel Aviv University, Tel Aviv, Israel
(e-mail: nogaa@tau.ac.il)

²Department of Communication Systems Engineering, Ben Gurion University of the Negev,
Beer Sheva, Israel
(e-mail: avin@cse.bgu.ac.il, zvilo@cse.bgu.ac.il)

³Mathematical Institute, Academy of Sciences of the Czech Republic, Prague, Czech Republic
(e-mail: koucky@math.cas.cz)

⁴Department of Mathematics, The Weizmann Institute of Science, Rehovot, Israel
(e-mail: gady.kozma@weizmann.ac.il)

⁵Intel Corporation, Hudson, Massachusetts, USA
(e-mail: tuttle@acm.org)

Received 14 January 2010; revised 13 February 2011; first published online 7 April 2011

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.

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 at every step each player can traverse an edge of

[†] A preliminary version appeared in *Proc. SPAA 2008*, pp. 119–128.

[§] Research supported in part by an ERC Advanced Grant and by a USA–Israeli BSF grant.

[¶] Work partially supported by grant GA ČR 201/07/P276 and 201/05/0124.

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 neighbours 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 overheads, 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 [19, 35, 13, 34, 9, 25, 11].

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, 1]. The cover time is relevant to a wide range of algorithmic applications [25, 36, 27, 9], and methods of bounding the cover time of graphs have been thoroughly investigated [32, 2, 16, 14, 38, 31]. Several bounds on the cover time of particular classes of graphs have been obtained: see [16, 14, 28, 29, 17].

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 essentially 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 2 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 centre of the barbell, each bell is likely to attract at least

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	$n^2/2$	$n^2/2$	$\Theta(n^2)$	$\log(k)$	$\log(k)$
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	$\Theta(n \log n)$	$\Theta(n)$	$O(\log n \log \log n)$	$k, k < O(\log^{1-\epsilon} n)$	
Complete graph	$\Theta(n \log n)$	$\Theta(n)$	1	$k, k < n$	$k, k < n$
Expanders	$\Theta(n \log n)$	$\Theta(n)$	$\Theta(\log n)$	$k, k < n$	
E-R tandom graph ¹	$\Theta(n \log n)$	$\Theta(n)$	$O(\log n)$	$k, k < O(\log^{1-\epsilon} n)$	

¹ With p larger than the connectivity threshold.

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 centre, 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 (*i.e.*, of logarithmic order). 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 , maximized over all pairs of 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 3.2 and 3.3). 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 $k = O(\log n)$. 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 3.7), 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 [15, 12, 24]), 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 analysing 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 pseudo-random walks to identify such subsets. These walks either start from all the vertices of G or from a suitably chosen sample of its vertices. Deterministic algorithms concerned with the amount of used space [33, 8] use pseudo-random walks starting from all the vertices of G . Parallel randomized algorithms, e.g., [30, 26], 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 [15, 12, 24]. Algorithms in this area choose first a random set of representatives and then perform short random walks to discover connectivity between the representatives. Part of the analysis in [15] by Broder, Karlin, Raghavan and Upfal calculates the expected number of steps needed to cover the whole graph. Indeed, Broder, Karlin, Raghavan and Upfal 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 $O\left(\frac{m^2 \log^3 n}{k^2}\right)$, where m is the number of edges and n is the number of vertices of the graph [15]. Barnes and Feige in [12, 24] 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 4.8 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 3.7 gives a bound of $O((nt_m \log^2 n)/k)$ 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.

Since the conference version of the current paper [7], several papers have been published on many random walks. Cooper and Frieze [18] obtained tight bounds for d -regular

random graphs, sharpening our estimate for the speed-up for expanders, provided they satisfy certain additional (typical) properties. Elsässer and Sauerwald [21] showed that for many graphs, such as the d -dimensional tori, the speed-up cannot be more than k . Efremenko and Reingold [20] proved that, when starting the walks at the worst vertices, the hitting time cannot improve by more than an $O(k)$ factor and the cover time cannot be improved by more than $\min\{k \log n, k^2\}$.

2. Preliminaries

We begin with a quick review of the asymptotic notation 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 $\lim_{n \rightarrow \infty} f(n)/g(n) = 0$, and
- $f(n) = \omega(g(n))$ if $\lim_{n \rightarrow \infty} f(n)/g(n) = \infty$.

We let \ln denote the natural logarithm and let \log denote the logarithm to base 2.

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 neighbours of v , and let $d(v) = |N(v)|$ be the degree of v . A d -regular graph is a graph in which every node has degree d .

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 of moving in the next step to u is $Q_{vu} = \frac{1}{d(v)}$ for $(v, u) \in E$ and 0 otherwise.

For a graph G and a vertex i in G , let $\tau_i(G)$ denote the time taken by a simple random walk starting at i to visit all nodes in G . Formally $\tau_i = \min\{t : \{X_i(0), \dots, X_i(t)\} = V\}$ and clearly this is a random variable which denotes a stopping time. Let $C_i = \mathbb{E}[\tau_i]$ 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 [32, 2, 16, 14, 38, 5], although much less is known about the variance of the cover time (i.e., formally $\text{Var}(\tau_i)$). 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 [22, 23].

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

Theorem 2.1 (Matthews’ theorem [32]). *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 the upper bound is not always tight: for example in the line we have $C(G) = h_{\max}$. Since we are mostly interested in upper bounds on cover time, when referring to Matthews' bound we will typically mean the upper bound $C(G) \leq h_{\max} \cdot H_n$.

For an integer $t > 0$, a graph G and its vertices u and v , let $p_{u,v}^t$ 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 the *mixing time* t_m of G we mean the smallest integer $t > 0$ such that, for all vertices u in G , $\sum_v |p_{u,v}^t - \pi(v)| < 1/e$.

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_i^k 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 $C_i^k = \mathbb{E}[\tau_i^k]$ be the expected cover time for k walks starting from i . For a graph G , let $C^k(G) = \max_i C_i^k(G)$ 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 2.2. 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, $S^k(G) = \frac{C(G)}{C^k(G)}$.

We will typically think of k as a function of the size of the graph. 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 3.1. For $k \leq n$ and for a graph G that is either a complete graph or an expander on n vertices, 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 place a stronger upper bound on 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 $O(\log n)$. Our proof depends on a generalization of Matthews' bound for multiple random walks: $C^k(G) \leq \frac{e+o(1)}{k} \cdot h_{\max} \cdot H_n$ (see Theorem 4.2). Since Matthews' bound is known to be tight for the complete graph, expanders [16], d -dimensional grids for $d \geq 2$ [16], d -regular

balanced trees for $d \geq 2$ [37], Erdős–Rényi random graphs [17], and random geometric graphs [10] (in the last two cases, for a choice of parameters that guarantee connectivity with high probability), the following result shows that $k = O(\log n)$ random walks yield a linear speed-up for a large class of interesting and useful graphs.

Theorem 3.2. *If $C(G) = \Theta(h_{\max} \log n)$, then $S^k(G) = \Omega(k)$ for all $k = O(\log n)$.*

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 3.3. *If $g(n) = \frac{C(G)}{h_{\max}} \rightarrow \infty$ and $k = O(g^{1-\epsilon}(n))$ for some $\epsilon > 0$, 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 a strong upper bound on 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 $\Theta(\log k)$.

Theorem 3.4. *For all $k < e^{n/4}$, 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 that 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 centre of the barbell graph, the speed-up is *exponential* in k (this is not the case when starting at other nodes of the graph).

Theorem 3.5. *For a barbell graph B_n on n vertices (see Section 7 for a definition), if v_c is the centre of the barbell then $C_{v_c} = \Theta(n^2)$, but $C_{v_c}^k = O(n)$ for $k = \Theta(\log n)$.*

For the second question, proving a linear speed-up in general does indeed require bounding of 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 3.4, we can show that there may be a full spectrum of speed-up behaviours even for a single graph.

Theorem 3.6. *Let G be a two-dimensional $\sqrt{n} \times \sqrt{n}$ toroidal grid (for which Matthews’ bound is tight).*

- (1) For $k = O(\log n)$, the speed-up is $S^k(G) = \Omega(k)$.
 (2) For $k = \Omega(\log^3 n)$ 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 3.3, and 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 3.7. *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 $S^k = \Omega\left(\frac{k}{t_m \ln n}\right)$*

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 3.8. *For any graph G and any $k \geq 1$, $S^k(G) = O(k)$.*

Conjecture 3.9. *For any graph G and any $n \geq k \geq 1$, $S^k(G) = \Omega(\log k)$.*

Indeed, as a partial progress toward settling our conjecture, Efremenko and Reingold [20] showed that the speed-up is at most $\min\{k \log n, k^2\}$.

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, so we begin by stating this simple example from folklore for later use:

Lemma 4.1. *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 mother decides to help her k children to collect the coupons. Each time she buys a cereal and gets a coupon she gives it to the next-in-turn child in a round-robin fashion (i.e., child $i \bmod k$ gets the coupon from step i). Clearly, in expectation, after C visits to the grocery store, the mother has got all the different coupons. Note that each child had his own independent coupon collecting process, and each has the same number of coupons (plus or minus one). \square

We now show a linear speed-up in a much larger class of graphs, as long as $k = O(\log n)$. 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 4.2 (Baby Matthew Theorem). *If G is a graph on n vertices and $k = O(\log n)$, then*

$$C^k(G) = O\left(\frac{h_{\max} \cdot H_n}{k}\right).$$

Moreover, if $k = o(\log n)$ then

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

Proof. Let the starting vertex u of the k -walk be chosen. Fix any other vertex v in the graph G . Recall that for any two vertices u', v' in G , $h(u', v') \leq h_{\max}$. Thus, by Markov's inequality,

$$\mathbb{P}[\text{a random walk of length } eh_{\max} \text{ starting from } u \text{ does not hit } v] \leq 1/e.$$

Hence for any integer $r > 1$, the probability that a random walk of length erh_{\max} does not visit v is at most $1/e^r$. (We can view the walk as r independent trials to visit v .) Thus the probability that a random k -walk of length erh_{\max} starting from u does not visit v is at most $1/e^{kr}$. Set $r = \lceil (\ln n + 2 \ln \ln n)/k \rceil$. Then the probability that a random k -walk of length erh_{\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 $C^k(G) \leq erh_{\max} + C(G)/\ln^2 n \leq O(h_{\max} H_n/k)$. When $k = o(\log n)$, the loss from rounding r to an integer goes asymptotically to zero so we obtain the stronger bound. The theorem follows. □

When Matthews' bound is tight, we have $C(G) = \Theta(h_{\max} \log n)$, and the linear speed-up is an immediate corollary of Theorem 4.2.

Theorem 3.2. *If $C(G) = \Theta(h_{\max} \log n)$, then $S^k(G) = \Omega(k)$ for all $k = O(\log n)$.*

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 4.3. *For any graph G of size n large enough, any k and any function $f(n) \in \omega(1)$,*

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

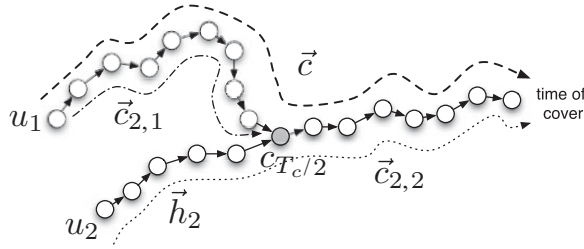


Figure 1. Example for the proof of Lemma 4.4 with $k = 2$. The second walk (dotted line) crosses the last vertex (grey vertex) of the first walk (dot-dashed line) and continues, generating a long walk (dashed line) that covers the graph.

We now prove Theorem 4.3. Our main technical tool, conceptually different from our previous proofs, is the following lemma.

Lemma 4.4. *Let G be a graph and let u_1, \dots, 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 $\lceil T_c/k \rceil + \ell T_h$ starting from vertices u_1, \dots, u_k covers G with probability at least $p_c(1 - (k - 1) \cdot (1 - p_h)^\ell)$.*

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

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

$$p = \mathbb{P} \left[\bigvee_{\vec{c}, \vec{h}_2, \dots, \vec{h}_k} \vec{c}_{k,1} \sqsubseteq X_1 \ \& \ \vec{h}_2 \circ \vec{c}_{k,2} \sqsubseteq X_2 \ \& \ \dots \ \& \ \vec{h}_k \circ \vec{c}_{k,k} \sqsubseteq X_k \right],$$

where \vec{c} is taken from the set of all sequences of vertices from G corresponding to walks of length $k \cdot \lceil T_c/k \rceil$ on G that start in u_1 and cover 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 $c_{(i-1)\lceil T_c/k \rceil}$ for the first time only at their end. Figure 1 illustrates the case where $k = 2$. A walk X_1 starts at u_1 and follows the sequence of vertices $\vec{c}_{2,1}$ that ends at vertex $c_{T_c/2}$. In parallel, walk X_2 starts at u_2 , and after less than ℓT_h steps (sequence \vec{h}_2) it reaches for the first time vertex $c_{T_c/2}$. It then follows the sequence $\vec{c}_{2,2}$. The joint sequence $\vec{c} = \vec{c}_{2,1} \circ \vec{c}_{2,2}$ covers the graph.

It is easy to verify that all the events in the union are disjoint. Hence,

$$\begin{aligned}
 p &= \sum_{\vec{c}, \vec{h}_2, \dots, \vec{h}_k} \mathbb{P}[\vec{c}_{k,1} \subseteq X_1 \ \& \ \vec{h}_2 \circ \vec{c}_{k,2} \subseteq X_2 \ \& \ \dots \ \& \ \vec{h}_k \circ \vec{c}_{k,k} \subseteq X_k] \\
 &= \sum_{\vec{c}, \vec{h}_2, \dots, \vec{h}_k} \mathbb{P}[\vec{c} \subseteq X_1 \ \& \ \vec{h}_2 \subseteq X_2 \ \& \ \dots \ \& \ \vec{h}_k \subseteq X_k] \\
 &= \sum_{\vec{c}, \vec{h}_2, \dots, \vec{h}_k} \mathbb{P}[\vec{c} \subseteq X_1] \cdot \mathbb{P}[\vec{h}_2 \subseteq X_2] \cdots \mathbb{P}[\vec{h}_k \subseteq X_k] \\
 &= \sum_{\vec{c}} \mathbb{P}[\vec{c} \subseteq X_1] \cdot \sum_{\vec{h}_2} \mathbb{P}[\vec{h}_2 \subseteq X_2] \cdots \sum_{\vec{h}_k} \mathbb{P}[\vec{h}_k \subseteq X_k],
 \end{aligned}$$

where the third equality follows from the independence of the walks. By our assumption,

$$\sum_{\vec{c}} \mathbb{P}[\vec{c} \subseteq X_1] \geq p_c.$$

Since $(1 - a)(1 - b) \geq (1 - a - b)$ for $0 \leq a, b \leq 1$, to conclude the lemma it suffices to argue that

$$\sum_{\vec{h}_i} \mathbb{P}[\vec{h}_i \subseteq X_i] \geq 1 - (1 - p_h)^\ell, \quad \text{for all } i.$$

Notice that

$$\sum_{\vec{h}_i} \mathbb{P}[\vec{h}_i \subseteq X_i] = \mathbb{P}[\text{a random walk of length } \ell T_h \text{ starting from } u_i \text{ visits } c_{(i-1) \cdot \lceil T_c/k \rceil}].$$

Since a random walk of length T_h fails to visit $c_{(i-1) \cdot \lceil T_c/k \rceil}$ with probability at most $1 - p_h$ regardless of its starting vertex, a random walk of length ℓT_h fails to visit $c_{(i-1) \cdot \lceil T_c/k \rceil}$ 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 [3].

Theorem 4.5 ([3]). *For the simple random walk on G , starting at i , if $C_i/h_{\max} \rightarrow \infty$ then $\tau_i/C_i \xrightarrow{p} 1$.*

Equipped with the proper tools we are ready to prove Theorem 4.3.

Proof of Theorem 4.3. If the conditions of Theorem 4.5 do not hold, then the cover time and hitting time are of the same order and Theorem 4.3 gives a trivial (not tight) upper bound. Assume the conditions of Theorem 4.5 hold. Theorem 4.5 implies that $\mathbb{P}[\tau_u/C_u > 1 + \delta_n] \leq \epsilon_n$, where $\delta_n, \epsilon_n \rightarrow 0$ as the size of the graph goes to infinity. Thus

$$\mathbb{P}[\text{a random walk of length } (1 + o(1)) \cdot C \text{ covers } G] \geq 1 - o(1).$$

By Markov’s bound, for a fixed vertex v of the graph,

$$\mathbb{P}[\text{a random walk of length } 2h_{\max} \text{ visits vertex } v] \geq 1/2.$$

If we set $\ell = \log k + \omega(1)$, then Lemma 4.4 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 $j \cdot L$ does not cover G with probability at most $[o(1)]^j$, so the cover time of G can be bounded by

$$L \sum_{j=1}^{\infty} j \cdot [o(1)]^{j-1} = L \cdot \frac{1}{1 - o(1)} = L \cdot (1 + o(1)). \quad \square$$

It follows from Theorem 4.3 that 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, a simple calculation shows that this happens when $\log k \cdot h_{\max} \leq C/k$ or $k \log k \leq C/h_{\max}$, which happens, for example, 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 4.6. *If $C = \Theta(h_{\max} \log n)$ and $k = O(\log^{1-\epsilon} n)$ for some $\epsilon > 0$, then*

$$C^k = \frac{C}{k} + o\left(\frac{C}{k}\right) \quad \text{and} \quad S^k(G) \geq k - o(k).$$

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 3.3. *If $g(n) = \frac{C(G)}{h_{\max}} \rightarrow \infty$ and $k = O(g^{1-\epsilon}(n))$ for some $\epsilon > 0$, then*

$$C^k(G) = \frac{C}{k} + o\left(\frac{C}{k}\right) \quad \text{and} \quad S^k(G) \geq k - o(k).$$

Proof. Set $f(n) \in \omega(1)$ in Theorem 4.3 to be $\log(g(n))$, and the claim follows. □

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 $k \leq n$. A graph is an *expander* if any set X of at most half of its vertices has at least $c|X|$ neighbours outside the set, where $c > 0$ is a constant bounded away from zero.

Theorem 4.7. *If G is an expander, then the speed-up $S^k(G) = \Omega(k)$ for $k \leq n$.*

An (n, d, λ) -graph is a d -regular graph G on n vertices so that the absolute value of every non-trivial eigenvalue of the adjacency matrix of G is at most λ . Here we consider d

to be constant. It is well known (see [6]) that a d -regular graph on n vertices (with a loop in every vertex) is an expander 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 4.8. *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 , $s < i \leq 2s$, let Y_i be the indicator random variable whose value is 1 if and only if 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–Schwarz,

$$\mathbb{P}[Y > 0] = \sum_{j>0} \mathbb{P}[Y = j] \geq \frac{\left(\sum_{j>0} j \mathbb{P}[Y = j]\right)^2}{\sum_{j>0} j^2 \mathbb{P}[Y = j]} = \frac{(\mathbb{E}(Y))^2}{\mathbb{E}(Y^2)}. \tag{4.1}$$

By linearity of expectation $\mathbb{E}(Y) = \sum_{i=s+1}^{2s} \mathbb{E}(Y_i)$. The expectation of Y_i is the probability that the walk visits v at step i . This is precisely the value of the coordinate corresponding to v in the vector $A^i z$, 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 $A^i z_2$ satisfies $\|A^i z_2\| \leq \left(\frac{\lambda}{d}\right)^i$, we conclude, by the definition of s , that each coordinate of $A^i z$ deviates from $1/n$ by at most $\frac{1}{2n}$.

It thus follows that

$$\mathbb{E}(Y) \geq \frac{s}{2n}. \tag{4.2}$$

By linearity of expectation

$$\mathbb{E}(Y^2) = \sum_{i=s+1}^{2s} \mathbb{E}(Y_i) + 2 \sum_{s < i < j \leq 2s} \mathbb{E}(Y_i Y_j).$$

Note that $\mathbb{E}(Y_i Y_j)$ is precisely the probability that the walk visits v at step i and 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

$$\mathbb{E}(Y^2) \leq \mathbb{E}(Y) + 2 \sum_{i=s+1}^{2s} \mathbb{E}(Y_i) \left(\frac{s}{n} + \sum_{r>0} (\lambda/d)^r \right) \leq \mathbb{E}(Y) \left[1 + \frac{2s}{n} + 2 \frac{\lambda}{d-\lambda} \right]. \tag{4.3}$$

Plugging the estimates (4.2) and (4.3) into (4.1), we conclude that

$$\mathbb{P}[Y > 0] \geq \frac{(\mathbb{E}(Y))^2}{\mathbb{E}(Y)[1 + 2s/n + 2\lambda/(d - \lambda)]} \geq \frac{s/(2n)}{1 + 2s/n + 2b} = \frac{s}{2n + 4s + 4bn}.$$

This completes the proof. □

Corollary 4.9. *Let G be an (n, d, λ) -graph and define $s = \frac{\ln(2n)}{\log(d/\lambda)}$, $b = \frac{\lambda}{d-\lambda}$. Suppose $n \geq 2s$, 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 $\frac{t}{2s}$ subwalks, each of length $2s$. By Lemma 4.8, for each of these subwalks, the probability it covers v is at least $\frac{s}{2n+4s+4bn} \geq \frac{s}{4(b+1)n}$. Note that this estimate holds for each specific subwalk, even after we expose all previous subwalks, since, 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 subwalks are exposed. It follows that the probability that v is not covered is at most

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

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 $t + \frac{1}{n} O(n \ln n) \leq O(t)$, for $k \leq n$.

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 $\frac{16(b+1)n \ln n}{k} > 2s = 2 \frac{\ln(2n)}{\log(d/\lambda)}$ 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 $k \leq n$.

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 3.7. *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 $S^k = \Omega\left(\frac{k}{t_m \ln n}\right)$*

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 $O\left(\frac{t_m n \ln^2 n}{k}\right)$. 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 $\lfloor i/k \rfloor$.

Define the random variables $Y_i = X_{\lfloor i/k \rfloor k \cdot 6t_m \ln n + (i \bmod k)}$. Hence, the Y_i 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 total variation distance of Y'_i from the uniform distribution on G is at most $2(1/e)^{6 \ln n} \leq 2/n^6$ [4, Chapter 4, Lemma 5]. In particular, for any vertex v of G , $|\mathbb{P}[Y'_i = v] - 1/n| \leq 1/n^6$.

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

$$(1/n - 1/n^6)^\ell \leq \mathbb{P}[Y'_k Y'_{k+1} \cdots Y'_{k+\ell-1} = v_1 \cdots v_\ell] \leq (1/n + 1/n^6)^\ell.$$

Hence,

$$1/n^\ell \cdot (1 - 1/n^2) \leq \mathbb{P}[Y'_k Y'_{k+1} \cdots Y'_{k+\ell-1} = v_1 \cdots v_\ell] \leq 1/n^\ell \cdot (1 + 2/n^2).$$

One can easily show (see the proof of Theorem 7.1) 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, the distribution of $Y'_k Y'_{k+1} \cdots Y'_{k+\ell-1}$, for $1 < \ell \leq n^3$, is close to a distribution of a random walk on a clique. Hence, the probability that $Y'_k, Y'_{k+1}, \dots, Y'_{k-1+10n \ln n}$ does not cover all the vertices of G is at most $(1/n^9 + 2/n^{11})$. Furthermore, unless $Y'_k, Y'_{k+1}, \dots, Y'_{k-1+10n \ln n}$ does not hit all the vertices of G , we can bound the expected cover time of G by

$$(6t_m \ln n) \cdot C^k(K_n) \cdot (1 + 2/n^2) = O\left(\frac{t_m \cdot n \ln^2 n}{k}\right).$$

If $Y'_k, Y'_{k+1}, \dots, Y'_{k-1+10n \ln n}$ does not hit all the vertices of G , we can bound the cover time of G by the trivial bound $O(n^3)$. Since $C(G) = \Omega(n \ln n)$, the claim follows. □

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 or almost 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 3.4. *For any integer n and $k < e^{n/4}$, 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 6.1. *Let $s > 1$ and $k \geq 1$ be such that $C^k \leq n^2/s$ for a cycle of length n . Then $k \geq e^{s/16}/8$.*

Proof. Assume that $C^k \leq n^2/s$ and we will prove that $k \geq e^{s/16}/8$. Pick an arbitrary vertex v of the cycle. Clearly, the cover time starting from the vertex v is $C^k_v \leq n^2/s$. Let a

random variable τ_v^k be the cover time of a random k -walk starting from v . By Markov's inequality, $\mathbb{P}[\tau_v^k \geq 2n^2/s] \leq 1/2$. 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 at time at most $2n^2/s$, then there is $1 \leq t \leq 2n^2/s$ 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 at least $1/2$ 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 by, since the probability of going to the left is the same as the probability of going to the right. By Chernoff's bound,

$$\begin{aligned} &\mathbb{P}[\text{the number of steps to the left and to the right of a walk differs by at least} \\ &n/2 \text{ at time } 2n^2/s] \\ &\leq 2e^{-\frac{sn^2}{16n^2}} \leq 2e^{-s/16}. \end{aligned}$$

Hence, the probability that a particular walk reaches the vertex $v_{n/2}$ during $2n^2/s$ steps is at most $4e^{-s/16}$.

Thus,

$$\mathbb{P}[\text{there exists a walk that reaches } v_{n/2} \text{ at time at most } 2n^2/s] \leq 4k \cdot e^{-s/16}.$$

Since this probability must be at least $1/2$, we conclude that $\frac{e^{s/16}}{8} \leq k$. □

It follows from the previous lemma that $S^k(L_n) = O(\log(k))$; the next lemma shows that $S^k(L_n) = \Omega(\log(k))$.

Lemma 6.2. *Let k be large enough and let n be an integer. If $k \leq e^{n/4}$, then $C^k \leq 2n^2/\ln k$ for a cycle of length n .*

To prove this lemma we need the following folklore statement.

Proposition 6.3. *For every real number $c \geq 2$ and every even integer $n \geq 16c^2$,*

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

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's bound. The lower bound can be derived as follows:

$$\mathbb{P}[(c-1)\sqrt{n} \leq X - n/2 \leq c\sqrt{n}] = \sum_{k=(c-1)\sqrt{n}}^{c\sqrt{n}} \mathbb{P}[X - n/2 = k].$$

For any k ,

$$\mathbb{P}[X - n/2 = k] = \binom{n}{n/2+k} / 2^n.$$

We will compare $\binom{n}{n/2+k}$ with the central binomial coefficient $\binom{n}{n/2}$:

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

We upper-bound this ratio as follows:

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

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

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

Thus

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

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

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

The proposition follows. □

Proof of Lemma 6.2. To prove that $C^k \leq \frac{2n^2}{\ln k}$, 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 has traversed around the whole cycle. Note that $n/2 = \sqrt{\ell}(c-1)$. By the previous proposition,

$$\begin{aligned} &\mathbb{P}[\text{a single walk makes at least } \ell/2 + n/2 \text{ steps to the right during a random walk} \\ &\text{of length } \ell] \\ &\geq e^{-3c^2-4} \geq 1/k, \end{aligned}$$

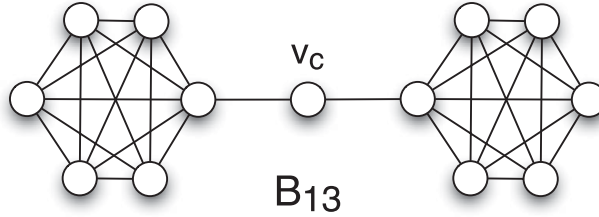


Figure 2. Example of a barbell graph B_{13} ; v_c is the centre of the barbell.

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 $(1 - 1/k)^k < 1/e$. Thus

$$C^k \leq \sum_{i=0}^{\infty} \frac{1}{e^i} \ell = e\ell / (e - 1) \leq 2n^2 / \ln k,$$

for k large enough. □

Lemma 6.1 also implies the following claim.

Theorem 6.4. *Let $G_{n,d}$ be a d -dimensional toroidal grid on $n^{1/d} \times n^{1/d} \times \dots \times n^{1/d}$ vertices, $d \geq 2$. For any k , $C^k(G_{n,d}) \geq \Omega(n^{2/d} / \log k)$.*

Proof. We prove the claim for $d = 2$. The other cases are analogous. Consider the random k -walk on a $\sqrt{n} \times \sqrt{n}$ toroidal grid. 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 at each step we make a step to the left with probability $1/4$, a 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 6.1. (Note that the steps at which we stay at the same vertex can only increase the cover time.) □

Corollary 6.5. *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 3.2 implies Theorem 3.6.

7. Exponential speed-up

In Definition 2.2 we formally defined the speed-up from the worst-case perspective, but as it turns out, on some graphs the speed-up can be exponential in k for some choice of the starting point (*i.e.*, not worst-case). 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 2). The vertex on that path is called the *centre* 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 centre of B_n . We show the following theorem (in which we make no attempt to optimize the absolute constants).

Theorem 7.1. *Let $n > 1$ be an odd integer, let v_c be the centre of B_n , and put $k = 20 \ln n$. The expected cover time starting from v_c satisfies $C_{v_c}^k = O(n)$.*

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 occurs.

- $\mathcal{E}1$ 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 $2 \ln n$ tokens return to the centre.
- $\mathcal{E}3$ One of the bells is not covered within the first $10n$ steps.

If none of the above events occurs 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 $2C^{2 \ln n}(K_m)$. So if C is the expected cover time of B_n by a random 1-walk, then

$$C_{v_c}^k \leq 2C^{2 \ln n}(K_m) + \mathbb{P}[\mathcal{E}1] \cdot C + \mathbb{P}[\mathcal{E}2 \cup \mathcal{E}3] \cdot (10n + C). \tag{7.1}$$

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

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

for n large enough. A single token returns to the centre 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 $2 \ln n$ tokens return to the centre 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

$$m \left(1 - \frac{1}{m} \right)^{20n \ln n} \leq me^{-10 \ln n} < 1/n^5.$$

Now, since $C = O(n^2)$ and $C^{2 \ln n}(K_m) = O(n)$ (by Lemma 4.1), we get $C_{v_c}^k = O(n)$. □

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 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 centre of the barbell achieved an exponential speed-up, but perhaps the speed-up is limited to k if we start at other nodes (see also [20]). 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 that this is possible on any graph.

Another source of open problems is to consider more general classes of graphs. In other words, 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?

References

- [1] Aldous, D. J. (1983) On the time taken by random walks on finite groups to visit every state. *Z. Wahrsch. Verw. Gebiete* **62** 361–374.
- [2] Aldous, D. J. (1989) Lower bounds for covering times for reversible Markov chains and random walks on graphs. *J. Theoret. Probab.* **2** 91–100.
- [3] Aldous, D. J. (1991) Threshold limits for cover times. *J. Theoret. Probab.* **4** 197–211.
- [4] Aldous, D. and Fill, J. (1999) Reversible Markov chains and random walks on graphs. Unpublished. <http://stat-www.berkeley.edu/users/aldous/RWG/book.html>.
- [5] Aleliunas, R., Karp, R. M., Lipton, R. J., Lovász, L. and Rackoff, C. (1979) 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, pp. 218–223.
- [6] Alon, N. (1986) Eigenvalues and expanders. *Combinatorica* **6** 83–96.
- [7] Alon, N., Avin, C., Koucký, M., Kozma, G., Lotker, Z. and Tuttle, M. R. (2008) Many random walks are faster than one. In *SPAA 2008: Proc. 20th Annual ACM Symposium on Parallel Algorithms and Architectures*, pp. 119–128.
- [8] Armoni, R., Ta-Shma, A., Wigderson, A. and Zhou, S. (2000) A $(\log n)^{4/3}$ space algorithm for (s, t) connectivity in undirected graphs. *J. Assoc. Comput. Mach.* **47** 294–311.
- [9] Avin, C. and Brito, C. (2004) Efficient and robust query processing in dynamic environments using random walk techniques. In *Proc. Third International Symposium on Information Processing in Sensor Networks*, pp. 277–286.
- [10] Avin, C. and Ercal, G. (2005) On the cover time of random geometric graphs. In *Proc. Automata, Languages and Programming, 32nd International Colloquium, ICALP05*, pp. 677–689.
- [11] Bar-Yossef, Z., Friedman, R. and Kliot, G. (2006) RaWMS: Random walk based lightweight membership service for wireless ad hoc network. In *MobiHoc '06: Proc. Seventh ACM International Symposium on Mobile ad hoc Networking and Computing* (New York 2006), ACM Press, pp. 238–249.
- [12] Barnes, G. and Feige, U. (1997) A spectrum of time–space tradeoffs for undirected s – t connectivity. *J. Comput. System Sci.* **54** 305–316.
- [13] Braginsky, D. and Estrin, D. (2002) Rumor routing algorithm for sensor networks. In *Proc. First ACM Internat. Workshop on Wireless Sensor Networks and Applications*, ACM Press, pp. 22–31.

- [14] Broder, A. and Karlin, A. (1989) Bounds on the cover time. *J. Theoret. Probab.* **2** 101–120.
- [15] Broder, A., Karlin, A., Raghavan, P. and Upfal, E. (1989) Trading space for time in undirected s - t connectivity. In *Proc. ACM Symp. Theory of Computing*, pp. 543–549.
- [16] Chandra, A. K., Raghavan, P., Ruzzo, W. L. and Smolensky, R. (1989) The electrical resistance of a graph captures its commute and cover times. In *Proc. 21st Annual ACM Symposium on Theory of Computing*, ACM Press, pp. 574–586.
- [17] Cooper, C. and Frieze, A. (2003) The cover time of sparse random graphs. In *Proc. 14th Annual ACM–SIAM Symposium on Discrete Algorithms: SODA-03* (Baltimore 2003), ACM Press, pp. 140–147.
- [18] Cooper, C., Frieze, A. M. and Radzik, T. (2009) Multiple random walks and interacting particle systems. In *ICALP 2* (S. Albers, A. Marchetti-Spaccamela, Y. Matias, S. E. Nikolettseas and W. Thomas, eds), Vol. 5556 of *Lecture Notes in Computer Science*, Springer, pp. 399–410.
- [19] Dolev, S., Schiller, E. and Welch, J. L. (2006) Random walk for self-stabilizing group communication in ad hoc networks. *IEEE Trans. Mob. Comput.* **5** 893–905.
- [20] Efremenko, K. and Reingold, O. (2009) How well do random walks parallelize? In *APPROX–RANDOM 2009* (I. Dinur, K. Jansen, J. Naor and J. D. P. Rolim, eds), Vol. 5687 of *Lecture Notes in Computer Science*, Springer, pp. 476–489.
- [21] Elsässer, R. and Sauerwald, T. (2009) Tight bounds for the cover time of multiple random walks. In *ICALP 1* (S. Albers, A. Marchetti-Spaccamela, Y. Matias, S. E. Nikolettseas and W. Thomas, eds), Vol. 5555 of *Lecture Notes in Computer Science*, Springer, pp. 415–426.
- [22] Feige, U. (1995) A tight upper bound on the cover time for random walks on graphs. *Random Struct. Alg.* **6** 51–54.
- [23] Feige, U. (1995) A tight lower bound on the cover time for random walks on graphs. *Random Struct. Alg.* **6** 433–438.
- [24] Feige, U. (1996) Short random walks on graphs. *SIAM J. Discrete Math.* **1** 19–28.
- [25] Gkantsidis, C., Mihail, M. and Saberi, A. (2006) Random walks in peer-to-peer networks: Algorithms and evaluation. *Perform. Eval.* **63** 241–263.
- [26] Halperin, S. and Zwick, U. (1996) An optimal randomized logarithmic time connectivity algorithm for the EREW PRAM. *J. Comput. System Sci.* **53** 395–416.
- [27] Jerrum, M. and Sinclair, A. (1997) The Markov chain Monte Carlo method: An approach to approximate counting and integration. In *Approximations for NP-hard Problems* (D. Hochbaum, ed.), PWS Publishing, Boston, pp. 482–520.
- [28] Jonasson, J. (1998) On the cover time for random walks on random graphs. *Combin. Probab. Comput.* **7** 265–279.
- [29] Jonasson, J. and Schramm, O. (2000) On the cover time of planar graphs. *Electron. Commun. Probab.* **5** 85–90.
- [30] Karger, D. R., Nisan, N. and Parnas, M. (1999) Fast connected components algorithms for the EREW PRAM. *SIAM J. Comput.* **28** 1021–1034.
- [31] Lovász, L. (1996) 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 Mathematical Society, Budapest, pp. 353–397.
- [32] Matthews, P. (1988) Covering problems for Brownian motion on spheres. *Ann. Probab.* **16** 189–199.
- [33] Nisan, N., Szemerédi, E. and Wigderson, A. (1992) Undirected connectivity in $O(\log^{1.5} n)$ space. In *Proc. 33rd Annual Symposium on Foundations of Computer Science*, pp. 24–29.
- [34] Sadagopan, N., Krishnamachari, B. and Helmy, A. (2005) ACQUIRE: Active query forwarding in sensor networks. *J. Ad Hoc Networks* **3** 91–113.
- [35] Servetto, S. D. and Barrenechea, G. (2002) Constrained random walks on random graphs: Routing algorithms for large scale wireless sensor networks. In *Proc. First ACM Internat. Workshop on Wireless Sensor Networks and Applications*, ACM Press, pp. 12–21.

- [36] Wagner, I. A., Lindenbaum, M. and Bruckstein, A. M. (1998) Robotic exploration, Brownian motion and electrical resistance. In *RANDOM'98*, Vol. 1518 of *Lecture Notes in Computer Science*, Springer, pp. 116–130.
- [37] Zuckerman, D. (1989) Covering times of random walks on bounded degree trees and other graphs. *J. Theoret. Probab.* **2** 147–157.
- [38] Zuckerman, D. (1990) A technique for lower bounding the cover time. In *Proc. 22nd Annual ACM Symposium on Theory of Computing*, ACM Press, pp. 254–259.