The Cover Time of Deterministic Random Walks

Tobias Friedrich[1] and Thomas Sauerwald[2]

[1] Max-Planck-Institut für Informatik, Campus E1.4, 66123 Saarbrücken, Germany
[2] Simon Fraser University, Burnaby B.C. V5A 1S6, Canada

Abstract The rotor router model is a popular deterministic analogue of a random walk on a graph. Instead of moving to a random neighbor, the neighbors are served in a fixed order. We examine how fast this "deterministic random walk" covers all vertices (or all edges). We present general techniques to derive upper bounds for the vertex and edge cover time and derive matching lower bounds for several important graph classes. Depending on the topology, the deterministic random walk can be asymptotically faster, slower or equally fast compared to the classical random walk.

1 Introduction

We examine the cover time of a simple deterministic process known under various names such as "rotor router model" or "Propp machine." It can be viewed as an attempt to derandomize random walks on graphs $G = (V, E)$. In the model each vertex $x \in V$ is equipped with a "rotor" together with a fixed sequence of the neighbors of x called "rotor sequence." While a particle (chip, coin, ...) performing a random walk leaves a vertex in a random direction, the deterministic random walk always goes in the direction the rotor is pointing. After a particle is sent, the rotor is updated to the next position of its rotor sequence. We examine how fast this model covers all vertices and/or edges, when one particle starts a walk from an arbitrary vertex.

1.1 Deterministic random walks

The idea of rotor routing appeared independently several times in the literature. First under the name "Eulerian walker" [28], then as "edge ant walk" [33] and later as "whirling tour" [16]. Around the same time it was also popularized by James Propp and analyzed by Cooper and Spencer [10] who called it the "Propp machine." Later the term "deterministic random walk" was established in Doerr *et al.* [11, 14]. For brevity, we omit the "random" and just refer to "deterministic walk."

Cooper and Spencer [10] showed the following remarkable similarity between the expectation of a random walk and a deterministic walk with cyclic rotor sequences: If an (almost) arbitrary distribution of particles is placed on the vertices of an infinite grid $\mathbb{Z}^d$ and does a simultaneous walk in the deterministic walk model, then at all times and on each vertex, the number of particles deviates from the expected number the standard random walk would have gotten there, by at most a constant. This constant is precisely known for the cases $d = 1$ [11] and $d = 2$ [14]. It is further known that there is no such constant for infinite trees [12]. Levine and Peres [24, 25] also extensively studied a related model called internal diffusion-limited aggregation for deterministic walks.

As in these works, our aim is to understand random walk and their deterministic counterpart from a theoretical viewpoint. However, we would like to mention that the rotor router mechanism also led to improvements in applications. With a random initial rotor direction, the quasirandom rumor spreading protocol broadcasts faster in some networks than its random counterpart [15], A similar idea is used in quasirandom load balancing [20].

We consider our model of a deterministic walk based on rotor routing to be a simple and canonic derandomization of a random walk which is not tailored for search problems. On the other hand, there is a vast literature on local deterministic agents/robots/ants patrolling or covering all vertices or edges of a graph (*e.g.* [21, 23, 30, 32, 33]). For instance, Cooper *et al.* [9] studied a model where the walk uses adjacent edges which have been traversed the smallest number of times. However, all of these models are more specialized and require additional counters/identifiers/markers/pebbles on the vertices or edges of the explored graph.

1.2 Cover time of random walks

In his survey, Lovász [26] mentions three important measures of a random walk: cover time, hitting time, and mixing time. These three (especially the first two) are closely related. Here we will mainly concentrate on the *cover time*, which is the expected number of steps to visit every node. The study of the cover time of random walks on graphs was initiated in 1979. Motivated by the space-complexity of the s–t-connectivity problem, Aleliunas *et al.* [2] showed that the cover time is upper bounded by $\mathcal{O}(|V|\,|E|)$ for any graph. For regular graphs, Feige [18] gave an improved upper bound of $\mathcal{O}(|V|^2)$ for the cover time. Broder and Karlin [4] proved several bounds which rely on the spectral gap of the transition matrix. Their bounds imply that the cover time on a regular expander graph is $\Theta(|V| \log |V|)$. In addition, many papers are devoted to the study of the cover time on special graphs such as hypercubes [1], random graphs [7], random regular graphs [6], random geometric graphs [8] etc. A general lower bound of $(1 - o(1))\,|V| \ln |V|$ for any graph was shown by Feige [17].

A natural variant of the cover time is the so-called *edge cover time*, which measures the expected number of steps to traverse all edges. Amongst other results, Zuckerman [35, 36] proved that the edge cover time of general graphs is at least $\Omega(|E| \log |E|)$ and at most $\mathcal{O}(|V|\,|E|)$.

1.3 Cover time of deterministic walks (our results)

For the case of a cyclic rotor sequence the edge cover time is known to be $\Theta(|E|\ \mathrm{diam}(G))$ (see Yanovski *et al.* [34] for the upper and Bampas *et al.* [3] for

Graph class G	Vertex cover time $\mathsf{VC}(G)$ of the random walk		Vertex cover time $\widetilde{\mathsf{VC}}(G)$ of the deterministic walk	
k-ary tree, $k = \mathcal{O}(1)$	$\Theta(n \log^2 n)$	[36, Cor. 9]	$\Theta(n \log n)$	(Thm. 4.2 and 3.13)
star	$\Theta(n \log n)$	[36, Cor. 9]	$\Theta(n)$	(Thm. 4.1)
cycle	$\Theta(n^2)$	[26, Ex. 1]	$\Theta(n^2)$	(Thm. 4.3 and 3.11)
lollipop graph	$\Theta(n^3)$	[26, Thm. 2.1]	$\Theta(n^3)$	(Thm. 4.4 and 3.14)
expander	$\Theta(n \log n)$	[4, Cor. 6], [31]	$\Theta(n \log n)$	(Thm. 4.5, Cor. 3.7)
two-dim. torus	$\Theta(n \log^2 n)$	[36, Thm. 4], [5, Thm. 6.1]	$\Theta(n^{1.5})$	(Thm. 4.6 and 3.11)
d-dim. torus ($d \geqslant 3$)	$\Theta(n \log n)$	[36, Cor. 12], [5, Thm. 6.1]	$\mathcal{O}(n^{1+1/d})$	(Thm. 3.11)
hypercube	$\Theta(n \log n)$	[1, p. 372], [27, Sec. 5.2]	$\Theta(n \log^2 n)$	(Thm. 4.7 and 3.12)
complete	$\Theta(n \log n)$	[26, Ex. 1]	$\Theta(n^2)$	(Thm. 4.1 and 3.10)

Table 1. Comparison of the vertex cover time of random and deterministic walk on different graphs ($n = |V|$).

the lower bound). It is further known that there are rotor sequences such that the edge cover time is precisely $|E|$ [28]. We allow arbitrary rotor sequences and present three techniques to upper bound the edge cover time based on the local divergence (Thm. 3.5), expansion of the graph (Thm. 3.6), and a corresponding flow problem (Thm. 3.9). With these general theorems it is easy to prove upper bounds for expanders, complete graphs, torus graphs, hypercubes, k-ary trees and lollipop graphs. In addition we show in Section 4 that these bounds can be matched by very canonical rotor sequences. Though a general lower bound of $\Omega(|E| \operatorname{diam}(G))$ was shown in [3], we believe that the study of these canonical rotor sequences is of independent interest. Unfortunately, all proofs had to be omitted to meet the page limit. They will be given in the full version of the paper.

It is not our aim to prove superiority of the deterministic walk, but it is instructive to compare our results for the vertex and edge cover time with the respective bounds of the random walk. Tables 1 and 2 group the graphs in three classes depending whether random or deterministic walk is faster. In spite of the strong adversary (as the order of the rotors is completely arbitrary), the deterministic walk is surprisingly efficient. It is known that the edge cover time of random walks can be asymptotically larger than its vertex cover time. Somewhat unexpectedly, this is not the case for the deterministic walk. To highlight this issue, let us consider hypercubes and complete graph. For these graphs, the vertex cover time of the deterministic walk is larger while the edge cover time is smaller (complete graph) or equal (hypercube) compared to the random walk.

2 Models and Preliminaries

2.1 Random Walks

We consider weighted random walks on finite connected graphs $G = (V, E)$. For this, we assign every pair of vertices $u, v \in V$ a weight $c(u, v) \in \mathbb{N}_0$ (rational weights can be handled by scaling) such that $c(u, v) = c(v, u) > 0$ if

Graph class G	Edge cover time $\mathsf{EC}(G)$ of the random walk	Edge cover time $\widetilde{\mathsf{EC}}(G)$ of the deterministic walk
k-ary tree, $k = \mathcal{O}(1)$	$\Theta(n \log^2 n)$ [36, Cor. 9]	$\Theta(n \log n)$ (Thm. 4.2 and 3.13)
star	$\Theta(n \log n)$ [36, Cor. 9]	$\Theta(n)$ (Thm. 4.1)
complete	$\Theta(n^2 \log n)$ [35, 36]	$\Theta(n^2)$ (Thm. 4.1 and 3.10)
expander	$\Theta(n \log n)$ [35, 36]	$\Theta(n \log n)$ (Thm. 4.5, Cor. 3.7)
cycle	$\Theta(n^2)$ [26, Ex. 1]	$\Theta(n^2)$ (Thm. 4.3 and 3.11)
lollipop graph	$\Theta(n^3)$ [26, Thm. 2.1], [35, Lem. 2]	$\Theta(n^3)$ (Thm. 4.4 and 3.14)
hypercube	$\Theta(n \log^2 n)$ [35, 36]	$\Theta(n \log^2 n)$ (Thm. 4.7 and 3.12)
two-dim. torus	$\Theta(n \log^2 n)$ [35, 36]	$\Theta(n^{1.5})$ (Thm. 4.6 and 3.11)
d-dim. torus ($d \geqslant 3$)	$\Theta(n \log n)$ [35, 36]	$\mathcal{O}(n^{1+1/d})$ (Thm. 3.11)

Table 2. Comparison of the edge cover time of random and deterministic walk on different graphs ($n = |V|$).

$\{u, v\} \in E$ and $c(u, v) = c(v, u) = 0$ otherwise. This defines transition probabilities $\mathbf{P}_{u,v} := c(u, v)/c(u)$ with $c(u) := \sum_{w \in V} c(u, w)$. So, whenever a random walk is at a vertex u it moves to a vertex v in the next step with probability $\mathbf{P}_{u,v}$. Moreover, note that for all $u, v \in V$, $c(u, v) = c(v, u)$ while $\mathbf{P}_{u,v} \neq \mathbf{P}_{v,u}$ in general. This defines a time-reversible, irreducible, finite Markov chain $X_0, X_1, \ldots$ with transition matrix $\mathbf{P}$. The t-step probabilities of the walk can be obtained by taking the t-th power of $\mathbf{P}^t$. In what follows, we prefer to use the term weighted random walk instead of Markov chain to emphasize the limitation to rational transition probabilities.

It is intuitively clear that a random walk with large weights $c(u, v)$ is harder to approximate deterministically with a simple rotor sequence. To measure this, we use $c_{\max} := \max_{u,v \in V} c(u, v)$. An important special case is the *unweighted random walk* with $c(u, v) \in \{0, 1\}$ for all $u, v \in V$ on a simple graph. In this case, $\mathbf{P}_{u,v} = 1/\deg(u)$ for all $\{u, v\} \in E$, and $c_{\max} = 1$. Our general results hold for weighted (random) walks. However, the derived bounds for specific graphs are only stated for unweighted walks. With *random walk* we mean unweighted random walk and if a random walk is allowed to be weighted we will emphasize this. For weighted and unweighted random walks we define for a graph G,

- cover time: $\mathsf{VC}(G) = \max_{u \in V} \mathbf{E}\left[\min\left\{t \geqslant 0 \colon \bigcup_{\ell=0}^{t} \{X_\ell\} = V\right\} \mid X_0 = u\right]$,
- edge cover time:
  $\mathsf{EC}(G) = \max_{u \in V} \mathbf{E}\left[\min\left\{t \geqslant 0 \colon \bigcup_{\ell=1}^{t} \{X_{\ell-1}, X_\ell\} = E\right\} \mid X_0 = u\right]$.

The (edge) cover time of a graph class $\mathcal{G}$ is the maximum of the (edge) cover times of all graphs of the graph class. Observe that $\mathsf{VC}(\mathcal{G}) \leqslant \mathsf{EC}(G)$ for all graphs G. For vertices $u, v \in V$ we further define

- (expected) hitting time: $\mathsf{H}(u, v) = \mathbf{E}\left[\min\{t \geqslant 0 \colon X_t = v\} \mid X_0 = u\right]$,
- stationary distribution: $\pi_u = c(u)/\sum_{w \in V} c(w)$.

2.2 Deterministic Random Walks

We define weighted deterministic random walks (or short: weighted deterministic walks) based on rotor routers as introduced by Holroyd and Propp [22].

For a weighted random walk, we define the corresponding weighted deterministic walk as follows. We use a tilde ($\widetilde{\ }$) to mark variables related to the deterministic walk. To each vertex u we assign a rotor sequence $\widetilde{s}(u) = (\widetilde{s}(u,1), \widetilde{s}(u,2), \ldots, \widetilde{s}(u, \widetilde{d}(u))) \in V^{\widetilde{d}(u)}$ of arbitrary length $\widetilde{d}(u)$ such that the number of times a neighbor v occurs in the rotor sequence $\widetilde{s}(u)$ corresponds to the transition probability to go from u to v in the weighted random walk, that is, $\mathbf{P}_{u,v} = |\{i \in [\widetilde{d}(u)] \colon \widetilde{s}(u,i) = v\}|/\widetilde{d}(u)$ with $[\widetilde{d}(u)] := \{1, \ldots, \widetilde{d}(u)\}$. For a weighted random walk, $\widetilde{d}(u)$ is a multiple of the lowest common denominator of the transition probabilities from u to its neighbors. For the standard random walk, a corresponding canonical deterministic walk would be $\widetilde{d}(u) = \deg(u)$ and a permutation of the neighbors of u as rotor sequence $\widetilde{s}(u)$. As the length of the rotor sequences crucially influences the performance of a deterministic walk, we set $\widetilde{\kappa} := \max_{u \in V} \widetilde{d}(u)/\deg(u)$ (note that $\widetilde{\kappa} \geqslant 1$). The set V together with $\widetilde{s}(u)$ and $\widetilde{d}(u)$ for all $u \in V$ defines the deterministic walk, sometimes abbreviated $\mathbf{D}$. Note that every deterministic walk has a unique corresponding random walk while there are many deterministic walks corresponding to one random walk.

We also assign to each vertex u an integer $\widetilde{r}_t(u) \in [\widetilde{d}(u)]$ corresponding to a rotor at u pointing to $\widetilde{s}(u, \widetilde{r}_t(u))$ at step t. A *rotor configuration* C describes the rotor sequences $\widetilde{s}(u)$ and initial rotor directions $\widetilde{r}_0(u)$ for all vertices $u \in V$. At every time step t the walk moves from $\widetilde{x}_t$ in the direction of the current rotor of $\widetilde{x}_t$ and this rotor is incremented[3] to the next position according to the rotor sequence $\widetilde{s}(\widetilde{x}_t)$ of $\widetilde{x}_t$. More formally, for given $\widetilde{x}_t$ and $\widetilde{r}_t(\cdot)$ at time $t \geqslant 0$ we set $\widetilde{x}_{t+1} := s(\widetilde{x}_t, \widetilde{r}_t(\widetilde{x}_t))$, $\widetilde{r}_{t+1}(\widetilde{x}_t) := \widetilde{r}_t(\widetilde{x}_t) \bmod \widetilde{d}(\widetilde{x}_t) + 1$, and $\widetilde{r}_{t+1}(u) := \widetilde{r}_t(u)$ for all $u \neq \widetilde{x}_t$. Let $\mathcal{C}$ be the set of all possible rotor configurations (that is, $\widetilde{s}(u)$, $\widetilde{r}_0(u)$ for $u \in V$) of a corresponding deterministic walk for a fixed weighted random walk (and fixed rotor sequence length $\widetilde{d}(u)$ for each $u \in V$). Given a rotor configuration $C \in \mathcal{C}$ and an initial location $\widetilde{x}_0 \in V$, the vertices $\widetilde{x}_0, \widetilde{x}_1, \ldots \in V$ visited by a deterministic walk are completely determined. For deterministic walks we define for a graph G and vertices $u, v \in V$,

- deterministic cover time:
  $\widetilde{\mathsf{VC}}(G) = \max_{\widetilde{x}_0 \in V} \max_{C \in \mathcal{C}} \min \big\{ t \geqslant 0 \colon \bigcup_{\ell=0}^{t} \{\widetilde{x}_\ell\} = V \big\}$,
- deterministic edge cover time:
  $\widetilde{\mathsf{EC}}(G) = \max_{\widetilde{x}_0 \in V} \max_{C \in \mathcal{C}} \min \big\{ t \geqslant 0 \colon \bigcup_{\ell=1}^{t} \{\widetilde{x}_{\ell-1}, \widetilde{x}_\ell\} = E \big\}$,
- hitting time: $\widetilde{\mathsf{H}}(u,v) = \max_{C \in \mathcal{C}} \min \{ t \geqslant 0 \colon \widetilde{x}_t = u, \widetilde{x}_0 = v \}$.

Note that the definition of the deterministic cover time takes the *maximum* over all possible rotor configurations, while the cover time of a random walk takes the *expectation* over the random decisions. Also, $\widetilde{\mathsf{VC}}(G) \leqslant \widetilde{\mathsf{EC}}(G)$ for all graphs G. We further define for fixed configurations $C \in \mathcal{C}$, $\widetilde{x}_0$, and vertices $u, v \in V$,

- number of visits to vertex u: $\widetilde{N}_t(u) = \big| \{ 0 \leqslant \ell \leqslant t \colon \widetilde{x}_\ell = u \} \big|$,

[3] In this respect we slightly deviate from the model of Holroyd and Propp [22] who first increment the rotor and then move the chip, but this change is insignificant here.

- number of traversals of a directed edge $u \to v$: $\widetilde{N}_t(u \to v) = \big|\{1 \leqslant \ell \leqslant t \colon (\widetilde{x}_{\ell-1}, \widetilde{x}_\ell) = (u,v)\}\big|$.

2.3 Graph-Theoretic Notation

We consider finite, connected graphs $G = (V,E)$. Unless stated differently, $n := |V|$ is the number vertices and $m := |E|$ the number of undirected edges. By δ and Δ we denote the minimum and maximum degree of the graph, respectively. For a pair of vertices $u, v \in V$, we denote by $\mathrm{dist}(u,v)$ their distance, i.e., the length of a shortest path between them. For a vertex $u \in V$, let $\Gamma(u)$ denote the set of all neighbors of u. More generally, for any $k \geqslant 1$, $\Gamma^k(u)$ denotes the set of vertices v with $\mathrm{dist}(u,v) = k$. For any subsets $S, T \subseteq V$, $E(S)$ denotes the set of edges with one endpoint in S and $E(S,T)$ denotes the edges $\{u,v\}$ with $u \in S$ and $v \in T$. As a walk is something directed, we also have to argue about directed edges though our graph G is undirected. In slight abuse of notation, for $\{u,v\} \in E$ we might also write $(u,v) \in E$ or $(v,u) \in E$.

3 Upper Bounds on the Deterministic Cover Times

Very recently, Holroyd and Propp [22] proved that several natural quantities of the weighted deterministic walk as defined in Section 2.2 concentrate around the respective expected values of the corresponding weighted random walk. To state their result formally, we set for a vertex $v \in V$,

$$K(v) := \max_{u \in V} \mathsf{H}(u,v) + \frac{1}{2}\left(\frac{\widetilde{d}(v)}{\pi_v} + \sum_{i,j \in V} \widetilde{d}(i)\, \mathbf{P}_{i,j}\, |\mathsf{H}(i,v) - \mathsf{H}(j,v) - 1|\right).$$

Theorem 3.1 ([22, Thm. 4]) *For all weighted deterministic walks, all vertices $v \in V$, and all times t, $\big|\pi_v - \widetilde{N}_t(v)/t\big| \leqslant K(v)\,\pi_v/t$.*

Roughly speaking, Theorem 3.1 states that the proportion of time spent by the weighted deterministic walk concentrates around the stationary distribution for all configurations $C \in \mathcal{C}$ and all starting points $\widetilde{x}_0$. To quantify the hitting or cover time with Theorem 3.1, we choose $t = K(v) + 1$ to get $\widetilde{N}_t(v) > 0$. To get a bound for the edge cover time, we choose $t = 3K(v)$ and observe that then $\widetilde{N}_t(v) \geqslant 2\pi_v K(v) > \widetilde{d}(v)$. This already shows the following corollary.

Corollary 3.2 *For all weighted deterministic walks,*

$$\begin{aligned} \widetilde{\mathsf{H}}(u,v) &\leqslant K(v) + 1 && \textit{for all } u, v \in V, \\ \widetilde{\mathsf{VC}}(G) &\leqslant \max_{v \in V} K(v) + 1, \\ \widetilde{\mathsf{EC}}(G) &\leqslant 3 \max_{v \in V} K(v). \end{aligned}$$

One obvious question that arises from Theorem 3.1 and Corollary 3.2 is how to bound the value $K(v)$. While it is clear that $K(v)$ is polynomial in n (provided that $c_{\max}$ and $\widetilde{\kappa}$ are polynomially bounded), it is not clear how to get more precise upper bounds. A key tool to tackle the difference of hitting times in $K(v)$ is the following elementary lemma, where in case of a periodic walk the sum is taken as a Cesáro summation. The proof of this lemma and the proofs of all following results is omitted due to space limitations.

Lemma 3.3 *For all weighted random walks and all vertices* $i, j, v \in V$, $\sum_{t=0}^{\infty} \left(\mathbf{P}_{i,v}^t - \mathbf{P}_{j,v}^t\right) = \pi_v \left(\mathsf{H}(j,v) - \mathsf{H}(i,v)\right)$.

To analyze weighted random walks, we use the notion of local divergence which is an important quantity in the analysis of load balancing algorithms [19, 29].

Definition 3.4 *The local divergence of a weighted random walk is* $\Psi(\mathbf{P}) := \max_{v \in V} \Psi(\mathbf{P}, v)$, *where* $\Psi(\mathbf{P}, v)$ *is the local divergence w.r.t. to a vertex* $v \in V$ *defined as* $\Psi(\mathbf{P}, v) := \sum_{t=0}^{\infty} \sum_{\{i,j\} \in E} \left|\mathbf{P}_{i,v}^t - \mathbf{P}_{j,v}^t\right|$.

Using Corollary 3.2 and Lemma 3.3, we can prove the following bound on the hitting time of a deterministic walk.

Theorem 3.5 *For all deterministic walks and all vertices* $v \in V$,

$$K(v) \leqslant \max_{u \in V} \mathsf{H}(u,v) + \frac{\widetilde{\kappa}\, c_{\max}}{\pi_v} \Psi(\mathbf{P}, v) + 2m\, \widetilde{\kappa}\, c_{\max}.$$

Note that Theorem 3.5 is more general than just giving an upper bound for hitting and cover times via Corollary 3.2. It can be useful in the other directions, too. To give a specific example, we can apply the result of Theorem 4.7 that $\widetilde{\mathsf{EC}}(G) = \Omega(n \log^2 n)$ for hypercubes and $\max_{u,v} \mathsf{H}(u,v) = \mathcal{O}(n)$ (cf. [26]) to Theorem 3.5 and obtain a lower bound of $\Omega(n \log^2 n)$ on the local divergence of hypercubes.

3.1 Upper bound depending on the expansion

We now derive an upper bound for $\widetilde{\mathsf{EC}}(G)$ that depends on the expansion properties of G. Let $\lambda_2(\mathbf{P})$ be the second-largest eigenvalue in absolute value of $\mathbf{P}$.

Theorem 3.6 *For all graphs* G, $\widetilde{\mathsf{EC}}(G) = \mathcal{O}\big(\frac{\Delta}{\delta} \frac{n}{1-\lambda_2(\mathbf{P})} + n\, \widetilde{\kappa}\, \frac{\Delta}{\delta} \frac{\Delta \log n}{1-\lambda_2(\mathbf{P})}\big)$.

Here, we call a graph with constant maximum degree an expander graph, if $1/(1 - \lambda_2(\mathbf{P})) = \mathcal{O}(1)$ (equivalently, we have for all subsets $X \subseteq V, 1 \leqslant |X| \leqslant n/2$, $|E(X, X^c)| = \Omega(|X|)$ (cf. [13, Prop. 6])). Using Theorem 3.6, we immediately get the following upper bound on $\widetilde{\mathsf{EC}}(G)$ for expanders.

Corollary 3.7 *For all expander graphs,* $\widetilde{\mathsf{EC}}(G) = \mathcal{O}(\widetilde{\kappa}\, n \log n)$.

3.2 Upper bound by flows

We relate the edge cover time of the unweighted random walk to the optimal solution of the following flow problem.

Definition 3.8 *Consider the flow problem where a distinguished source node s sends a flow amount of 1 to each other node in the graph. Then $f_s(i,j)$ denotes the load transferred along edge $\{i,j\}$ (note $f_s(i,j) = -f_s(j,i)$) such that $\sum_{\{i,j\}\in E} f_s(i,j)^2$ is minimized.*

Theorem 3.9 *For all graphs G,*

$$\widetilde{\mathsf{EC}}(G) = \mathcal{O}\bigg(\frac{\Delta}{\delta}\max_{u,v\in V}\mathsf{H}(u,v) + \Delta\, n\,\widetilde{\kappa} + \widetilde{\kappa}\,\Delta \max_{s\in V}\sum_{\{i,j\}\in E}|f_s(i,j)|\bigg)$$

where f_s is the flow with source s according to Definition 3.8.

3.3 Upper bounds for common graphs

We now demonstrate how to apply above general results to obtain upper bounds for the edge cover time of the deterministic walk for many common graphs. As the general bounds Theorems 3.5, 3.6 and 3.9 all have a linear dependency on $\widetilde{\kappa}$, the following upper bounds can be also stated depending on $\widetilde{\kappa}$. However, for clarity we assume $\widetilde{\kappa} = \mathcal{O}(1)$ here.

Theorem 3.10 *For complete graphs, $\widetilde{\mathsf{EC}}(G) = \mathcal{O}(n^2)$.*

Theorem 3.11 *For d-dimensional torus graphs ($d \geqslant 1$ constant), $\widetilde{\mathsf{EC}}(G) = \mathcal{O}(n^{1+1/d})$.*

Theorem 3.12 *For hypercubes, $\widetilde{\mathsf{EC}}(G) = \mathcal{O}(n\log^2 n)$.*

Theorem 3.13 *For k-ary trees ($k \geqslant 2$ constant), $\widetilde{\mathsf{EC}}(G) = \mathcal{O}(n\log n)$.*

Theorem 3.14 *For lollipop graphs, $\widetilde{\mathsf{EC}}(G) = \mathcal{O}(n^3)$.*

4 Lower Bounds on the Deterministic Cover Time

We first give a general lower bound of $\Omega(m)$ on the deterministic cover time for all graphs. Afterwards, for all graphs examined in Section 3.3 for which this general bound is not tight we present stronger lower bounds which match their respective upper bounds.

Theorem 4.1 *For all graphs, $\widetilde{\mathsf{VC}}(G) \geqslant m - \delta$.*

Theorem 4.2 *For k-ary trees ($k \geqslant 2$ constant), $\widetilde{\mathsf{VC}}(G) = \Omega(n\log n)$.*

Theorem 4.3 *For cycles, $\widetilde{\mathsf{VC}}(G) = \Omega(n^2)$.*

Theorem 4.4 *For lollipop graphs, $\widetilde{\mathsf{VC}}(G) = \Omega(n^3)$.*

Theorem 4.5 *There are expander graphs with $\widetilde{\mathsf{VC}}(G) = \Omega(n\log n)$.*

Theorem 4.6 *For two-dimensional torus graphs, $\widetilde{\mathsf{VC}}(G) = \Omega(n^{3/2})$.*

Theorem 4.7 *For hypercubes, $\widetilde{\mathsf{VC}}(G) = \Omega(n\log^2 n)$.*

Bibliography

[1] D. Aldous. On the time taken by random walks on finite groups to visit every state. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, pages 361–374, 1983.

[2] R. Aleliunas, R. Karp, R. Lipton, L. Lovász, and C. Rackoff. Random walks, universal traversal sequences, and the complexity of maze problems. In *20th Annual IEEE Symposium on Foundations of Computer Science (FOCS '79)*, pages 218–223, 1979.

[3] E. Bampas, L. Gasieniec, N. Hanusse, D. Ilcinkas, R. Klasing, and A. Kosowski. Euler tour lock-in problem in the rotor-router model. In *International Symposium on Distributed Computing (DISC'09)*, volume 5805 of *LNCS*, pages 423–435. Springer, 2009.

[4] A. Broder and A. Karlin. Bounds on the cover time. *Journal of Theoretical Probability*, 2(1):101–120, 1989.

[5] A. Chandra, P. Raghavan, W. Ruzzo, R. Smolensky, and P. Tiwari. The electrical resistance of a graph captures its commute and cover times. *Computational Complexity*, 6(4):312–340, 1997.

[6] C. Cooper and A. Frieze. The cover time of random regular graphs. *SIAM Journal of Discrete Mathematics*, 18(4):728–740, 2005.

[7] C. Cooper and A. Frieze. The cover time of the giant component of a random graph. *Random Structures & Algorithms*, 32(4):401–439, 2008.

[8] C. Cooper and A. Frieze. The cover time of random geometric graphs. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '09)*, pages 48–57, 2009.

[9] C. Cooper, D. Ilcinkas, R. Klasing, and A. Kosowski. Derandomizing random walks in undirected graphs using locally fair exploration strategies. In *36th International Colloquium on Automata, Languages, and Programming (ICALP '09)*, pages 411–422, 2009.

[10] J. Cooper and J. Spencer. Simulating a random walk with constant error. *Combinatorics, Probability & Computing*, 15:815–822, 2006.

[11] J. Cooper, B. Doerr, J. Spencer, and G. Tardos. Deterministic random walks on the integers. *European Journal of Combinatorics*, 28(8):2072–2090, 2007.

[12] J. Cooper, B. Doerr, T. Friedrich, and J. Spencer. Deterministic random walks on regular trees. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '08)*, pages 766–772, 2008.

[13] P. Diaconis and L. Saloff-Coste. Comparison theorems for reversible Markov chains. *Annals of Applied Probability*, 3(3):696–730, 1993.

[14] B. Doerr and T. Friedrich. Deterministic random walks on the two-dimensional grid. *Combinatorics, Probability & Computing*, 18(1-2):123–144, 2009.

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

[16] I. Dumitriu, P. Tetali, and P. Winkler. On playing golf with two balls. *SIAM J. Discrete Math.*, 16(4):604–615, 2003.

[17] U. Feige. A tight lower bound for the cover time of random walks on graphs. *Random Structures & Algorithms*, 6(4):433–438, 1995.
[18] U. Feige. Collecting coupons on trees, and the cover time of random walks. *Computational Complexity*, 6(4):341–356, 1997.
[19] T. Friedrich and T. Sauerwald. Near-perfect load balancing by randomized rounding. In *41st Annual ACM Symposium on Theory of Computing (STOC '09)*, pages 121–130, 2009.
[20] T. Friedrich, M. Gairing, and T. Sauerwald. Quasirandom load balancing. In *21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '10)*, pages 1620–1629, 2010.
[21] L. Gasieniec, A. Pelc, T. Radzik, and X. Zhang. Tree exploration with logarithmic memory. In *18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '07)*, pages 585–594, 2007.
[22] A. E. Holroyd and J. Propp. Rotor walks and markov chains, 2009. arXiv:0904.4507.
[23] R. E. Korf. Real-time heuristic search. *Artif. Intell.*, 42(2-3):189–211, 1990.
[24] L. Levine and Y. Peres. Spherical asymptotics for the rotor-router model in $\mathbb{Z}^d$. *Indiana Univ. Math. J.*, 57:431–450, 2008.
[25] L. Levine and Y. Peres. Strong spherical asymptotics for rotor-router aggregation and the divisible sandpile. *Potential Analysis*, 30:1–27, 2009.
[26] L. Lovász. Random walks on graphs: A survey. *Combinatorics, Paul Erdös is Eighty*, 2:1–46, 1993.
[27] J. L. Palacios. Expected hitting and cover times of random walks on some special graphs. *Random Structures & Algorithms*, 5(1):173–182, 1994.
[28] V. B. Priezzhev, D. Dhar, A. Dhar, and S. Krishnamurthy. Eulerian walkers as a model of self-organized criticality. *Phys. Rev. Lett.*, 77:5079–5082, 1996.
[29] Y. Rabani, A. Sinclair, and R. Wanka. Local divergence of Markov chains and the analysis of iterative load balancing schemes. In *39th Annual IEEE Symposium on Foundations of Computer Science (FOCS '98)*, pages 694–705, 1998.
[30] O. Reingold. Undirected connectivity in log-space. *J. ACM*, 55(4), 2008.
[31] R. Rubinfeld. The cover time of a regular expander is $\mathcal{O}(n \log n)$. *Inf. Process. Lett.*, 35(1):49–51, 1990.
[32] I. A. Wagner, M. Lindenbaum, and A. M. Bruckstein. Smell as a computational resource - a lesson we can learn from the ant. In *Israeli Symposium on Theory of Computing and Systems (ISTCS '96)*, pages 219–230, 1996.
[33] I. A. Wagner, M. Lindenbaum, and A. M. Bruckstein. Distributed covering by ant-robots using evaporating traces. *IEEE Transactions on Robotics and Automation*, 15(5):918–933, 1999.
[34] V. Yanovski, I. A. Wagner, and A. M. Bruckstein. A distributed ant algorithm for efficiently patrolling a network. *Algorithmica*, 37(3):165–186, 2003.
[35] D. Zuckerman. On the time to traverse all edges of a graph. *Inf. Process. Lett.*, 38(6):335–337, 1991.
[36] D. Zuckerman. A technique for lower bounding the cover time. *SIAM Journal on Discrete Mathematics*, 5(1):81–87, 1992.