Better Bounds for Coalescing-Branching Random Walks

Michael Mitzenmacher [*]
Harvard University
michaelm@eecs.harvard.edu

Rajmohan Rajaraman [†]
Northeastern University
rraj@ccs.neu.edu

Scott Roche [‡]
Northeastern University and
Akamai Technologies
str7@cornell.edu

ABSTRACT

Coalescing-branching random walks, or *cobra walks* for short, are a natural variant of random walks on graphs that can model the spread of disease through contacts or the spread of information in networks. In a k-cobra walk, at each time step a subset of the vertices are active; each active vertex chooses k random neighbors (sampled independently and uniformly with replacement) that become active at the next step, and these are the only active vertices at the next step. A natural quantity to study for cobra walks is the cover time, which corresponds to the expected time when all nodes have become infected or received the disseminated information.

In this work, we extend previous results for cobra walks in multiple ways. We show that the cover time for the 2-cobra walk on an n-vertex d-dimensional grid is $O(n^{1/d})$ (where the order notation hides constant factors that depend on d); previous work had shown the cover time was $O(n^{1/d} \cdot \text{polylog}(n))$. We show that the cover time for a 2-cobra walk on an n-vertex d-regular graph with conductance ϕ_G is $O(d^4\phi_G^{-2}\log^2 n)$, significantly generalizing a previous result that held only for expander graphs with sufficiently high expansion. And finally we show that the cover time for a 2-cobra walk on a graph with n vertices is always $O(n^{11/4}\log n)$; this is the first result showing that the bound of $\Theta(n^3)$ for the worst-case cover time for random walks can be beaten using 2-cobra walks.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Stochastic processes; Probabilistic Algorithms; G.2.2 [**Discrete Mathematics: Graph Theory**]: Graph algorithms

Keywords

Random Walks; Networks; Information Spreading; Cover Time; Epidemic Processes

1. INTRODUCTION

Random walks provide a fundamental mathematical model for many basic network processes. In disease models, transmission of a virus can be modeled by the virus moving according to a random walk on a graph representing a human contact network; computer viruses can be modeled similarly [18, 26, 13]. Variants of random walks can also be used for information dissemination, using message-passing protocols where a message is passed from neighbor to neighbor via a random walk [9, 17]. Such protocols require little state information and are robust to various types of faults, and are therefore useful in many distributed networks [4]. More generally, random walks provide a fundamental primitive for network algorithms for information propagation, search, routing, and load balancing.

In many of these settings, a central measure of interest is the cover time, the expected time for the random walk to cover all of the vertices of the underlying network. In disease models, this corresponds to the time until all vertices in the network have been exposed to the virus; in message-passing protocols, this corresponds to the time until all vertices have received the message.

Parallel random walks provide a natural generalization of standard random walks, with multiple random walks traversing the network simultaneously, and several papers have analyzed the performance of parallel random walks (as we describe in the related work section). A related variant, less well-studied and understood, are coalescing-branching random walks, or *cobra walks* for short [13]. In a cobra walk, at each time step, a subset of the vertices are active; typically in the initial state a single vertex would be active. At each time step, each active vertex chooses k random neighbors (sampled independently and uniformly with replacement) that become active at the next step. A vertex is active at step t if and only if it was chosen by an active vertex in the previous step. When $k > 1$, each walk branches at that step into multiple walks, but multiple walks then coalesce when they reach the same vertex at the same time. We refer to a cobra walk where at each time step a vertex chooses k active neighbors as a k-cobra walk for convenience. (One could further study variations where the branching varied based on the vertex or the time step, or was governed by a random distribution; we do not do that here.)

[*] This work was supported in part by NSF grants CNS-1228598, CCF-1320231, and CCF-1535795.

[†] Supported in part by NSF CCF-1422715, NSF CCF-1535929, and an ONR grant

[‡] Supported in part by NSF CCF-1216038 and NSF CCF-1422715.

SPAA '16, July 11 - 13, 2016, Pacific Grove, CA, USA

ISBN 978-1-4503-4210-0/16/07…$15.00

DOI: http://dx.doi.org/10.1145/2935764.2935791

As examples of cobra walks, in the message passing setting, a k-cobra walk corresponds to a network where a vertex may send k outgoing copies of the message to neighbors during a time step instead of just one. In disease networks, a cobra walk corresponds to an idealized process within the Susceptible Infected Susceptible model (or SIS model): in each time step, an infected agent infects k random neighbors and recovers, but can be infected again (including at the next time step).

In contrast to results in parallel random walks, where the number of walks is a parameter, in cobra walks the number of active vertices varies over time and its behavior depends significantly on the network. One might expect cobra walks to yield significant improvements in the cover time over standard random walks, based on their power to reproduce, even if limited by coalescence. The goal of our work is to formally and theoretically bound the performance of cobra walks, focusing on the cover time. While our results have potential applications to distributed protocols and disease models, as suggested above, we also believe that k-cobra walks are a natural mathematical model worthy of study in their own right.

1.1 Our Results and Techniques

We are motivated by the prior work [13], which obtained bounds on the cover time of cobra walks on trees, grids, and expanders. Our work pushes those results further, in several directions. Our primary results are the following:

- We show that the cover time for the 2-cobra walk on an n-vertex d-dimensional grid is $O(n^{1/d})$, where the constant in the order notation can depend on d. This improves on the previous bound of $O(n^{1/d} \cdot \mathrm{polylog}(n))$ [13]. With respect to n, our result is optimal.
- We show that the cover time for the 2-cobra walk for a d-regular graph with conductance ϕ_G is $O(\phi_G^{-2} \log^2 n)$. This generalizes a similar result in [13] for expander graphs with sufficiently high expansion. Our new result holds for any d-regular graph, and expresses the bound as a function of the conductance.
- We provide a result for general graphs, showing that the cover time for a 2-cobra walk is always $O(n^{11/4} \log n)$. For standard random walks, there are graphs where the cover time is $\Theta(n^3)$. This is the first result showing that cobra walks can beat the corresponding worst-case bound for random walks. We also establish an $O(n^{2-1/d} \log n)$ upper bound on the cover time for arbitrary d-regular graphs, again improving the tight quadratic bound for standard random walks.

Our main techniques involve making use of the parallelism inherent in cobra walks, and by thinking of cobra walks as a *union of biased random walks.* In some settings we can show the cobra walk goes through an initial phase that instantiates a large number of essentially parallel random walks, and then analyze the behavior of these random walks. Here we have to take care of the dependency challenges introduced by coalescing, as random walks can essentially disappear when several collide at a vertex. In other settings, we think of our cobra walk as being a single walk moving toward a specific vertex, and then eventually taking a union bound over all vertices. At each step, we can choose to follow the active vertex that moves toward the target vertex and discard the others. This approach simplifies the analysis by allowing us to focus on a single walk, where now the k choices correspond to a bias in the walk that we can model. The downside of such an analysis, however, is that it does not take full advantage of the power of parallelism inherent to cobra walks.

1.2 Background and Related Work

The cobra walk process has structural similarities with several other fairly diverse stochastic processes: branching processes, gossip protocols, and random walks (including parallel random walks, coalescing random walks, and other variants). Despite these commonalities, cobra walks resist being fully described by any of these other processes; furthermore, analysis techniques used for these other processes often have no clear use or power in analyzing cobra walks.

Branching and coalescing processes. Branching processes appear in many disciplines, from nuclear physics to population genetics, on various discrete and continuous structures [19, 21, 6]. Another related topic is the study of coalescing processes and voter models (see, for example [11]). Naturally, there is also work on processes that contain both branching and coalescing elements [3, 25], although unlike our work these analyses tend to operate in continuous time and on either restricted topologies or infinite spaces. The restriction of cobra walks to discrete time effectively disallows the use of differential-equation based analysis that often yields results in continuous-time processes (see, for example [18, 20, 26]).

Gossip and rumor-spreading mechanisms Gossip-based algorithms have been used successfully to design efficient distributed algorithms for a variety of problems in networks such as information dissemination, aggregate computation, constructing overlay topologies, and database synchronization (e.g., see [9] and the references therein). There are three major variants of gossip-based processes: push-based models, in which the members of the set of informed vertices each select a neighbor and inform that neighbor (if it is not already informed), pull-based models, in which uninformed vertices select neighbors and poll them for information, and push-pull, which is a combination of the first two.

Cobra walks bear the closest resemblance to push-based gossip models. Indeed, [17] show that the push process completes in every undirected graph in $O(n \log n)$ steps with high probability, and this bound has been conjectured to hold for cobra walks [13]. However, the similarity between the two is in many ways superficial. If we view a gossip process as a Markov chain on the state space of the set of all subsets of vertices (representing the sets of possible informed vertices), this Markov chain has a single absorbing state (assuming the graph is connected) in which every vertex is informed. On the other hand, performing a similar projection of a cobra walk onto a Markov chain of the 2^n possible subsets of vertices that could be active at any time, we see that there is no absorbing state and with the addition of self-loops, the chain can be made ergodic.

Random walks and parallel random walks. Cobra walks also resemble standard random walks and parallel variants. For simple random walks, the now classic work of Feige [15, 16] showed that the cover time on any graph lies between $\Theta(\log n)$ and $O(n^3)$. A formal model of biased random walks was introduced in [5] with the motivation of studying imperfect sources of randomness. Specifically, these biased walks allow a controller to fix, at each step of the walk, the next step with a small probability, with the aim of increasing the stationary probability at a target set of vertices. A variant of the biased walk of [5] plays a significant role in our analysis of cobra walks for general graphs.

Additional work has considered speeding up the cover time by modifying the underlying process. Adler et al [1] studied a process on the hypercube in which in each round a vertex is chosen uniformly at random and covered; if the chosen vertex was already covered, then an uncovered neighbor of the vertex is chosen uniformly at random and covered. For any d-regular graph, Dimitrov and Plaxton showed that a similar process achieves a cover time of $O(n + (n \log n)/d)$ [12]. For expander graphs, Berenbrink et al showed a simple variant of the standard random walk that achieves a linear (i.e., $O(n)$) cover time [7].

Parallel random walks, first studied in [8] for the special case where the starting vertices are drawn from the stationary distribution and in [2] for arbitrary starting vertices, also appear related to cobra walks. Nearly-tight results on the speedup of cover time as a function of the number of parallel walks have been obtained by [14] for several graph classes including the cycle, d-dimensional meshes, hypercube, and expanders. However, again the similarity is somewhat superficial. A parallel random walk with k independent walks can be mapped to anundirected random walk on a graph known as the tensor product. As such, much of the machinery of the analysis of simple random walks can be applied to the parallel case. Applying a similar approach to a cobra walk is not feasible, although for cobra walks one can convert the tensor product into a directed graph, changing the topology significantly. As such, generally the techniques that can be used for parallel random walks cannot be used directly for the cobra walk. Indeed, one can view the dependencies on the locations of the other active nodes in a cobra walk as a manifestation of this difference. Cobra walks suffer from the “time's arrow” effect: locally, most individual steps are reversible, but as a cobra walk expands, the likelihood that it will coalesce back to a single vertex grows exponentially unlikely. Despite these difficulties, the tensor product graph can be useful when studying the movement of a small number of pebbles in a cobra walk, and we make use of this technique in obtaining a general bound based on conductance.

2. PRELIMINARIES

Let G be a connected graph with vertex set V and edge set E, and let $|V| = n$, except for the case when we are analyzing the grid, in which case we let $V = [0, n]^d$. A k-coalescing-branching (k-cobra) walk is defined as follows: It starts at time $t = 0$ at an arbitrary vertex v, at which a pebble is placed. In the next and every subsequent time step, every pebble in G clones itself $k - 1$ times (so that there are now k indistinguishable pebbles at each vertex that originally had a pebble). Each pebble then independently selects a neighbor of its current vertex uniformly at random and moves to it. Once all pebbles have made their moves, the coalescing phase begins: if two or more pebbles are at the same vertex they coalesce into a single pebble, and the next round begins.

For time step t, S_t is the **active set**, which is the set of all vertices of G that have a pebble. Define the **cover time** of a cobra walk to be the maximum over all vertices v of expectation of the minimum time T at which all vertices have belonged to some S_t for $t \leq T$ when the cobra walk is started at v. We note that while our results are stated as bounds on the cover time, all of the results in this paper actually give bounds on the time to cover all the vertices in the graph with high probability, as is clear from the proofs. Hence we may also refer to the time at which all vertices have been covered, where the meaning is clear. The **hitting time** $H(u, v)$ is the expectation of the minimum time it takes for any pebble originating from a cobra walk that starts at u to reach v. The **maximum hitting time** $h_{\max}$ is $max_{u,v \in V} H(u, v)$.

We make use of an extension of Matthews' Theorem, which relates the cover time of a random walk to the maximum hitting time. The following theorem was proven in [13]:

THEOREM 1. *Let G be a connected graph on n vertices. Let W be a cobra walk on G starting at an arbitrary vertex. Then the cover time of W on G is bounded above by $O(h_{\max} \log n)$; in fact W covers all of G in $O(h_{\max} \log n)$ steps with high probability.*

Finally, we make use of a combinatorial property of the graph, the conductance. Define the conductance of a set $S \subseteq V$ as $\phi(S) = |\partial(S)|/vol(S)$, where $\partial(S) = \sum_{(u,v):u \in S, v \notin S} 1$ and $vol(S) = \sum_{u \in S} d(u)$. Then the conductance Φ_G of the graph is $\min_{S:vol(S) \leq vol(V)/2} \phi(S)$. For the purposes of this paper, we say that a d-regular graph is an ϵ-expander if the conductance of the graph is greater than or equal to ϵ.

3. TIGHT RESULTS FOR GRIDS

We show that the cover time for the d-dimensional grid using a 2-cobra walk $[0, N]^d$ is $O(N)$, where the order notation hides constant factors and other terms that depend on d; indeed, we show all vertices are covered in $O(N)$ steps with high probability. Previous work has shown that the cover time is $O(N\text{polylog}(N))$ [13]. We note that the results of [13] use a slightly different notation, working with n total nodes, or $[0, n^{1/d} - 1]^d$. We have opted to work over $[0, N]^d$ for convenience, with conversion factor $N = n^{1/d} - 1$. Our result is clearly tight in its dependence on N. Moreover, it shows that in some circumstances one can avoid using tools such as Matthews' Theorem, which had been used previously in this setting [13], and necessarily adds in an additional logarithmic factor in the number of vertices over the hitting time.

The case of $d = 2$ is simple and instructive; we sketch a proof, but do not go into full detail as we have a more detailed proof for the general case.

LEMMA 2. *The 2-cobra walk on $[0, N]^2$ has cover time $O(N)$.*

PROOF. Let v_0 be the starting vertex of the walk. We show that with high probability all vertices are reached within $O(N)$ steps, and the result follows, because if every vertex is hit within T steps with probability p, the expected time to cover all vertices is bounded above by T/p.

Let $v_1 = (x_1, y_1)$ be some other vertex on the grid. Let X_t be the Manhattan distance between the closest pebble of the cobra walk and v_1 after t steps, which we will refer to as time t. (All distances in this section will refer to Manhattan distances.) Let $u_t = (a_t, b_t)$ be some arbitrary vertex with a pebble of distance X_t from v_1 at time t. We show by cases that there is drift so that the expectation of X_t decreases linearly over time, even when at each step we pessimistically consider only the single vertex u_t and not other additional pebbles; it follows from standard results in random walks v_1 is reached after $O(N)$ steps with probability $1 - O(1/N^3)$. Hence all vertices are covered after $O(N)$ steps with probability $1 - O(1/N)$, and the result follows.

If $x_1 \neq a_t$ and $y_1 \neq b_t$, the probability that at least one of the two pebbles at u_t moves closer to v_1 is at least $1 - (1/2)^2 = 3/4$, since each pebble moves closer to v_1 with probability at least $1/2$. (It can be more if a pebble v_1 is on the boundary of a grid.) If either $x_1 = a_t$ or $y_1 = b_t$ but not both, the probability that at least one of the two pebbles at u_t moves closer to v_1 could be as small as $1 - (3/4)^2 = 7/16 < 1/2$, since each pebble could move closer to v_1 with probability only $1/4$. (This probability would be $1 - (2/3)^2 = 5/9$ if for example $x_1 = a_t = n$, but off the grid boundary this is not the case.) Over one step, then, the expected distance may be increasing. So we instead consider two steps. It is important to note that if the distance increases on the first of the two steps but (at least) one pebble moves so that $x_1 \neq a_t$ and $y_1 \neq b_t$ after the first step, it improves our probability that the distance decreases in the second step. We find taking cases, assuming that u_t is at least distance 2 from the boundary and from v_1, that X_t increases by 2 by with probability $\frac{1}{16}\frac{9}{16} + \frac{1}{2}\frac{1}{4} = \frac{41}{256}$. X_t decreases by 2 by with probability $\frac{7}{16}\frac{7}{16} = \frac{49}{256}$. (Similar (better) results can be shown when u_t is near a boundary.)

We therefore see that X_t has negative drift (except at $X_t = 1$, where the drift is slightly positive), and therefore the time to reach 0 can be shown to be $O(N)$ with probability $1 - O(1/N)$ as claimed. □

The analysis above suggests technical difficulties to overcome with a direct approach for general d; the behavior is slightly different at the boundaries of the grid (though one could always work on the toroidal grid), and when coordinates match in one or more dimensions, it makes analyzing the drift more difficult. The analysis makes clear that for any fixed d, there should be a large enough constant value of k so that the cover time for the k-cobra walk is $O(N)$ on the d-dimensional grid $[0, N]^d$. One just needs a large enough value of k so that the pebble nearest a target vertex drifts toward that vertex.

We actually show the stronger result that the 2-cobra walk has cover time $O(N)$ on the d-dimensional grid $[0, N]^d$, where the order notation hides constants that depend on d. We prove this below; we have not aimed to optimize the constant factors. The intuition for the proof is the following. If we look at the distance between the closest point on the cobra walk and our target vertex in any single dimension, it behaves like a biased random walk, with a bias toward 0. Hence, after $O(N)$ steps, each individual dimension has matched coordinates, with high probability. Indeed, if each chain was an independent biased random walk with constant bias (independent of d), after $O(dN)$ steps we would expect each independent walk to be near its stationary distribution, in which case each chain would be at 0 with some constant probability γ, and hence it would take roughly γ^d steps for all chains to be at 0 simultaneously. Sadly, as usual, the fact that the chains are not themselves independent causes significant technical challenges.

THEOREM 3. *The 2-cobra walk has cover time $O(N)$ on the d-dimensional grid $[0, N]^d$ for any constant d.*

PROOF. We break the proof into steps. As in Lemma 2, we consider the distance to a target vertex over all dimensions for some pebble generated by the 2-cobra walk. We pessimistically keep track of only a single pebble. Specifically, our state at time t can be defined as follows. Let $(z_{1,0}, z_{2,0}, \ldots, z_{d,0})$ be such that the distance from the initial pebble to the target vertex is $z_{i,0}$ for the ith grid dimension. More generally, assume there is some pebble at the tth step so that $(z_{1,t}, z_{2,t}, \ldots, z_{d,t})$ gives the distance from that pebble to the target vertex in each dimension. We update the $z_{i,t}$ values over time steps as follows. If our two choices of pebbles generated from that pebble move in the same dimension, we choose the pebble that moves closer to the target, if such a pebble exists. If our two choices of pebbles generated from that pebble move in different dimensions i and j, there are several cases. If $z_{i,t} = 0$ but $z_{j,t} \neq 0$, we choose the pebble that moves in dimension j. If $z_{i,t} = 0$ and $z_{j,t} = 0$, we choose the pebble randomly. If $z_{i,t} \neq 0$ and $z_{j,t} \neq 0$, if both choices of pebbles move closer or both move farther away from the target, we choose the pebble randomly; otherwise we choose the pebble that moves closer.

LEMMA 4. *In each dimension, we have that if $z_{i,t} \neq 0$, then $z_{i,t}$ changes in the next step with probability at least $1/(2d-1)$, and conditioned on the ith dimension being the value that changes, it decreases with probability at least $1/2 + 1/(8d-4)$. If $z_{i,t} = 0$, it increases in the next step with probability at most $2/(d+1)$.*

PROOF. This follows directly from the description above. We note that worst case with regard to the bias is when one dimension has $z_{i,t} \neq 0$ and $z_{j,t} \neq 0$ for $j \neq i$. In this case, the only bias that favors the ith dimension decreasing rather than increasing stems for both choices being in the ith dimension. With probability $2(d-1)/d^2$ the pebble is chosen to move in the ith dimension and some other dimension, in which case the pebble is equally likely to move closer or further to the target. With probability $1/d^2$ dimension i is chosen for both moves, in which case it moves closer with probability $3/4$. Hence, conditioned on $z_{i,t}$ changing, it increases with probability

$$\frac{(d-1)/d^2 + (3/4)/d^2}{2(d-1)/d^2 + 1/d^2} = \frac{d - 1/4}{2d-1} = \frac{1}{2} + \frac{1}{8d-4}$$

as claimed.

With regard to which dimension moves, we notice that when $z_{i,t} \neq 0$, the probability of moving is least when we are on the boundary in the ith dimension (and hence there is just one move in that dimension), and other dimensions are not. When $z_{i,t} = 0$, that dimension is most likely to move if all others are on the boundary. The above bounds reflect these cases. □

Our multi-dimensional biased random walk has a natural interpretation as a discrete time queueing system, where customers arrive and wait at a randomly chosen queue, where the arrival rate is slightly smaller than the departure rate (except when a queue is empty). In this setting, our question concerns the time until the system empties from a given starting state. Surprisingly, despite this connection, we could not find a statement corresponding to our desired result in the literature.

The following follows easily from the bias shown above.

Lemma 5. *In each individual dimension, the walk starting at $z_{i,0}$ with probability $1 - O(1/N^{d+1})$, $z_{i,t}$ hits 0 in $O(d^2N)$ steps.*

Proof. Let X be the number of steps taken in the ith dimension over the first $64d^2N$ steps. Then by Lemma 4, $\mathbb{E}[X] > 32dN$, and using a Chernoff bound (e.g., [24, Exercise 4.7])

$$\Pr(X \leq 16dN) \leq (e/2)^{-8dN}.$$

The expected difference between the number of steps that decrease $z_{i,t}$ and the number of steps that increase $z_{i,t}$ grows with X, so we pessimistically condition on $X \geq 16dN$. Suppose that in this case that 0 is never reached, so the bias remains in effect over all $16dN$ steps. Let Y be number of decreases in the first $16dN$ steps. Then $\mathbb{E}[Y] \geq 8dN + 2N$, and again using a Chernoff bound (e.g., [24, Exercise 4.7])

$$\Pr(Y \leq 8dN + N \mid X \geq 16dN) \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{-8dN+2N},$$

where $\delta = 1/(8d+2)$. Notice that if $Y \geq 8dN + N$ then in fact 0 was reached. Hence the total probability that 0 is not reached is bounded above by

$$\Pr(Y \leq 8dN+N \mid X \geq 16dN)+\Pr(X \leq 16dN) \leq (e/2)^{-8dN}$$

which is clearly $O(1/N^{d+1})$. □

Similarly, the following result is standard for biased random walks.

Lemma 6. *Once $z_{i,t}$ hits 0, with probability $1-O(1/N^{d+2})$, it will remain below $c_d \ln N$ for some constant c_d (depending on d) over the next $100d^2N^2$ steps.*

Proof. We may pessimistically assume that all N^2 steps are performed in the ith dimension. By the natural coupling we have that the probability that $z_{i,t}$ reaches $c_d \ln N$ in exactly k steps after it hits 0 is less than the probability in equilibrium that a biased random walk with probability $\frac{1}{2} + \frac{1}{8d-4}$ is at $c_d \ln N$. (Technically, the biased random walk doesn't have an equilibrium distribution, because of parity; it will be an even number of steps from its starting point after an even number of steps. We can add an arbitrarily small self-loop probability and increase the number of steps accordingly; we use N^2 steps and assume the appropriate equilibrium distribution for convenience.) The equilibrium distribution for this biased random walk, where π_j is the probability of being at j in equilibrium, is easily found by detailed balance equations, which yield

$$\pi_j = \frac{2}{4d-1}\left(\frac{4d-3}{4d-1}\right)^k.$$

As π_j is geometrically decreasing, for $j = c_d \ln n$ for some constant c_d we have π_k will be less than $1/(100d^2N^{d+4})$, and hence over the $100d^2N^2$ time steps we see $z_{i,t}$ never reaches $c_d \ln N$ with probability $1 - O(1/N^{d+2})$. □

We now use the following lemma, which in a slightly different form appears in Theorem 7 of [13]. We sketch the proof for completeness.

Lemma 7. *Starting from $(z_{1,t}, z_{2,t}, \ldots, z_{d,t})$, where each $z_{i,t}$ is at most $c_d \ln N$, with probability $\Omega(1/(\ln \ln N)^c)$ for some constant c, there is a $u = \alpha \ln n$ such that for some $k \leq u$, $z_{i,t+k} = 0$ for all $i = 1, \ldots, d$.*

Proof. The analysis is broken into phases. The first phase is of length $O(\ln N)$, the second phase is of length $O((\ln N)^{1/2})$, and the jth phase is of length $O((\ln N)^{1/2^{j-1}})$. All phases have the same basic structure, except for the last. Within each phase, there are d subphases. In the ith subphase, we assume that the ith coordinate $z_{i,t}$ moves according to a biased random walk as previously described; in all other phases, we pessimistically assume that it moves according to an unbiased random walk. Our goal is to show that in each phase, each dimension moves closer to zero; in particular, after the jth phase, all the $z_{i,t}$ are at $O((\ln N)^{1/2^j})$ with constant probability. This is shown using Chernoff bounds (see Theorem 7 of [13] for the corresponding calculation). It follows that after $O(\ln \ln \ln N)$ phases each $z_{i,t}$ will be bounded by a constant with probability $\Omega(1)^{O(\ln \ln \ln N)} = \Omega(1/(\ln \ln N)^c)$. At that point, there is a constant probability that the target vertex will be reached in a constant number of steps. □

Putting this all together now yields the theorem that the cover time for the 2-cobra random walk on $[0, N]^d$ is $O(N)$ where the order notation hides constant factors that can depend on d. Lemma 5 shows that for each dimension i, $z_{i,t}$ hits 0 within the first $64d^2N^2$ steps with high probability. Hence, by Lemma 6, after $64d^2N^2$ steps all $z_{i,t}$ are at most $c_d \ln N$ with high probability. We are therefore in a state where we reach the target vertex within an additional $O(\ln N)$ steps with probability $\Omega(1/(\ln \ln N)^c)$. If we have not reached the target vertex, however, we are still within $c_d \ln N$ distance in each dimension with high probability. That is, let $\mathcal{E}_1$ be the event that we did not reach the target vertex within the $\alpha \ln N$ steps of Lemma 7, and let $\mathcal{E}_2$ be the event that some $z_{i,t}$ is more than $c_d \ln n$ after those $\alpha \ln N$ steps. We have by Lemma 6 that this is a low probability event (at least up through an additional $36d^2N^2$ steps), so we can consider a polylogarithmic number of repeated trials of $\alpha \ln N$ steps. As each trial succeeds with probability $\Omega(1/(\ln \ln N)^c)$, we can conclude that we hit the target vertex with probability at least $1 - O(1/N^{d+1})$ within $O(N)$ steps. It follows via a union bound that with probability $1 - O(1/N)$ all vertices are hit within $O(N)$ steps, from which it readily follows that the cover time is $O(N)$.

We remark that this proof, while achieving $O(N)$ bounds, appears quite loose in the constant factors. As our proof does not directly take advantage of the large number of pebbles within the system, we believe the bounds could be tightened with respect to the dependence on d. Even with this bound, we expect there remains further work to fully understand the behavior of cobra walks on grids.

We also remark that the multi-step case analysis used in Lemma 2 can similarly be used to show that 2-cobra walks on k-ary trees have cover times that are proportional to the graph's diameter when $k = 2$ or $k = 3$. We conjecture that in fact the cover time for 2-cobra walks on k-ary trees is proportional to the diameter for every constant k, where the constant of proportionality may depend on k, similar to Theorem 3 for grids.

4. COVER TIME FOR A GRAPH WITH ARBITRARY CONDUCTANCE

In this section, we significantly improve and extend the results first developed in [13], which provided an $O(\log^2 n)$ bound for the cover time of a k-cobra walk on a d-regular expander with an expansion only achieved by graphs such as random $d-$regular expanders and Ramanujan expanders. Here, we provide the first known bound for the cover time of cobra walks on a d-regular graph of *arbitrary* conductance Φ. While the upper bound is not useful for graphs with very low conductance, there are a wide class of graphs beyond expanders for which this guarantees rapid coverage,, e.g. the hypercube, power-law graphs, and random geometric graphs. For this section and the rest of the paper, we work exclusively with 2-cobra walks, and use cobra walks to mean 2-cobra walks where the meaning is clear.

THEOREM 8. *Let G be a d-regular graph with conductance Φ_G. Then a cobra-walk starting at any vertex will cover G in $O(d^4\Phi_G^{-2}\log^2(n))$ rounds, with high probability.*

From this general bound, we have the following corollary, which improves the previous result of [13].

COROLLARY 9. *Let G be a bounded-degree d-regular ϵ-expander graph. Then a cobra walk starting at any vertex v will cover G in $O(\log^2(n))$ rounds with high probability when d is specifically bounded by a constant.*

Due to the extreme difficulty of analyzing the progress of a cobra walk explicitly, we follow [13] and analyze a process that, while conceptually similar to a cobra walk, has more structured rules which allow us to analyze walks taken by individual pebbles with only limited dependence on one another. Furthermore, this process stochastically dominates the cobra walk when starting from the same vertex with respect to the time to cover all of the vertices. Any upper bound on the cover time for this process therefore automatically applies to the cover time of a cobra walk as well.

This process, which we refer to as W_{alt}, can be defined as follows: We start with δn pebbles for some constant $\delta \leq 1/2$, distributed arbitrarily among the vertex set V, including the possibility for more than one pebble to to start at the same vertex. Furthermore, we assume that the pebbles have a total ordering, and that each pebble knows its position in the ordering. In this process, unlike in the cobra walk, no pebbles split or coalesce – the total number of pebbles is an invariant. Pebbles interact with one another according to two simple rules. For each time step:

1. If one or two pebbles are co-located in time and space: at vertex v at time t, each pebble chooses a random neighbor from $N(v)$ independently u.a.r and moves to that vertex.

2. If three or more pebbles are at v at time t, the two pebbles with the lowest order each pick a vertex independently from $N(v)$ u.a.r. and move to their chosen vertex. Label these vertices u, w (keeping in mind that $u = w$ is allowed). The remaining pebble(s) at v then each independently pick u with probability $1/2$ or w with probability $1/2$ and move to the vertex they have chosen.

If we are observing, for a single time step, a vertex v at which two or more pebbles (or zero, trivially) have landed, we would be unable to distinguish between a cobra walk and a W_{alt} process. On the other hand, if we observe a vertex v at which a single pebble has landed, we *would* be able to distinguish. In W_{alt}, in the next step, the pebble at v will act like a simple random walk and move to a single neighbor. On the other hand, in the cobra walk, there is some probability p two neighbors will receive a pebble from v, and probability $1 - p$ only one neighbor will. Thus, the active set of W_{alt} can be viewed as a (possibly proper) subset of the active set of a cobra walk when both are started from the same initial state. Therefore, at any future time t, the size of the active set of a cobra walk (viewed as a random variable) stochastically dominates the size of the active set of W_{alt}. We can then "invert" this argument to show that the cover time of W_{alt} stochastically dominates the cover time of the cobra walk. For a formal proof, we refer to the full paper [23].

Finally, for technical reasons, we make the W_{alt} process a *lazy* process. That is, at each step, with probability $1/2$ all pebbles remain in their same position. With probability $1/2$ a step proceeds with the probabilities described above. (Thus, to obtain the unconditioned probabilities of any particular action, we need to multiply the above probabilities through by $1/2$.)

We are now ready to prove Theorem 8. This proof builds on the machinery developed in [13, Theorem 16 and Lemma 17]. Here we will outline the proof technique, making note of the new ideas needed to provide a general bound on the cover time in terms of the conductance of the graph. We have refactored the full proof for completeness, and it is available in the full paper [23].

One significant difference with previous work is the following. The previous analysis consisted of two stages, with the first stage providing an exponential growth in the number of pebbles, but requiring the graph to have extremely high expansion. In contrast, we begin our analysis of W_{alt} with a large number of pebbles, *all of which are located at a single initial, arbitrary vertex v*, and compare this to a cobra walk that starts at v. In the first stage of Since the stochastic dominance of the cover time of W_{alt} over a cobra walk holds for all starting distributions of the pebbles on the vertices of G, it also therefore holds for the starting distribution in which all δn pebbles begin at the same vertex. This allows us to work exclusively with W_{alt} for the entire analysis, bypassing the analysis of the cobra walk as it grows from one active node to a linear number of active nodes and therefore dropping the requirement that G have extremely high expansion.

The second contribution of our new analysis is the derivation of bounds on the probability of coverage of vertices in terms of the conductance of the graph. As a consequence, we are able to derive bounds that hold for regular graphs with arbitrary values of d and arbitrary Φ_G.

Proof of Theorem 8: To prove the theorem, we break up W_{alt} into epochs of length s, where $s \in O(\Phi_G^m \log n)$, for m a constant to be defined later. We show that each vertex v has a constant probability of being hit by at least one pebble at the exact time step the epoch ends. We then go through $O(\log n)$ epochs to boost the probability v is covered to a sufficiently high probability and then take a union bound over every vertex to obtain the result.

We first prove that in a particular epoch vertex v will be hit by at least one pebble with constant probability. Define E_i to be the event that pebble i for $i \in 1, \ldots, \delta n$ covers an arbitrary vertex v at time s. Then the event that v is hit by at least one pebble is $\bigcup E_i$. Note that for any two pebbles i, j it is not safe to assume that E_i and E_j are independent, since i and j may cross paths during their walks and have their transition probabilities affected by the rules of W_{alt}. However, we can use a second-order inclusion-exclusion inequality to lower-bound the probability:

$$\Pr\left[\bigcup_i E_i\right] \geq \sum_i \Pr\left[E_i\right] - \sum_{i,j:i \neq j} \Pr\left[E_i \cup E_j\right]$$

Thus, we need to show that, for the r.h.s., the first quantity is a constant and the second quantity is smaller.

The first part is fairly straightforward and makes use of well-known results regarding simple random walks on graphs of a given conductance. We point the reader to the full version of the proof.

Next we establish an upper bound for $\Pr\left[E_i \cap E_j\right]$ using the joint (dependent) walks of pebbles i and j. Without loss of generality, assume that i has a lower order than j, and that if i and j are co-located in time and space, then not only is i the lower order, but j must behave like a third-or-greater priority pebble and choose the same vertex that i next hops to with probability $1/2$. (Note that under this assumption, the total probability that j moves to the same vertex as i is $1/2 + 1/2d$, a fact which will be important shortly).

We can view the random walks of i and j as a random walk over a graph with the topology of the tensor product graph $G \times G$. The tensor product graph has the Cartesian product $V(G) \times V(G)$ as its vertex set, and an edge set defined as follows: vertex $(u, u') \in V(G \times G)$ has an edge to $(v, v') \in V(G \times G)$ if and only if $(u, v), (u', v') \in E(G)$. The random walk we construct on $G \times G$ is slightly different than a simple random walk on the same graph: we make the edges directed, and we attach weights to them such that the walk on the directed graph $D(G \times G)$ is isomorphic to the movement of pebbles i,j in a W_{alt} on G.

We will show in the next Lemma that the walk on $D(G \times G)$ has a stationary distribution, that this stationary distribution is close to $1/n^2$, and that the walk converges rapidly to it. Thus, after s steps, the probability that i and j are at the same vertex in G is bounded from above by $2/(n^2+n) + 1/n^4$.

Once we have established bounds for $\Pr\left[E_i\right], \Pr\left[E_i \cap E_j\right]$, we can apply them to the full expression and get:

$$\begin{aligned}
\Pr\left[\bigcup E_i\right] &\geq \sum_i \Pr\left[E_i\right] - \frac{1}{2}\sum_{i \neq j} \Pr\left[E_j \cap E_i\right] \\
&\geq \delta n \frac{1}{2n} - \frac{1}{2}\binom{\delta n}{2}\left(\frac{2}{n^2+n} + \frac{1}{n^4}\right) \\
&\geq \frac{\delta}{2} - \frac{\delta}{4}(\delta n^2 - n)\left(\frac{2}{n^2+n} + \frac{1}{n^4}\right) \\
&\geq \frac{\delta}{2} - \frac{2\delta^2}{4}
\end{aligned}$$

Finally, to complete the proof, we have the following lemma:

Lemma 10. *Let G and s be defined as in Theorem 8. Let i, j be two pebbles walking on G according to the rules of W_{alt} If $E_i \cap E_j$ are defined as the event in which i and j are both located at (arbitrarily chosen) vertex v at time s, then* $\Pr\left[E_i \cap E_j\right] \leq \dfrac{2}{n^2+2} + \dfrac{1}{n^4}$.

Here we provide a brief sketch of the proof, full details of which are provided in the long version. We first create a directed graph $\mathcal{D}(G \times G) = \mathcal{D}$ through some minor modifications of the edge set of $G \times G$. These changes are such that the transition probability matrix of $\mathcal{D}$ is isomorphic to W_{alt} on G. We make use of the fact that $\mathcal{D}$ is an Eulerian digraph to calculate the stationary distribution π of the walk on $\mathcal{D}$. We then show that $\Phi_G = \Phi_{G \times G}$ and, making use of the work of [10], demonstrate through the use of the directed Cheeger inequalities that the second smallest eigenvalue of the normalized Laplacian of $\mathcal{D}$ is bounded from below by $O(\Phi_G^{-2})$. We then apply Theorem 7.3 of [10] to show that this walk converges in $O(\Phi^{-2} \log n)$ time to the stationary distribution of $\Theta(1/n^2)$ for each vertex in $\mathcal{D}$.

This concludes the proof of Theorem 8. It follows immediately that when d is a constant and G is any expander with conductance Φ_G bounded away from zero by a constant, a cobra walk will cover G in at most $O(\log^2 n)$ time. □

5. COVER TIME FOR GENERAL GRAPHS

In this section, we show that the cover time of 2-cobra walks for general graphs is $O(n^{11/4} \log n)$. For d-regular graphs, we obtain an improved bound of $O(n^{2-1/d} \log n)$. We establish the bounds on the cover time by proving that for any start vertex u and any target vertex v, the hitting time from u to v, denoted by $H(u, v)$, is $O(n^{11/4})$ for general graphs and $O(n^{2-1/d})$ for d-regular graphs. We then apply Matthews' Theorem (Theorem 1), which actually give a high probability result.

To analyze the cobra walk, we consider biased versions of the standard random walk; in each step, with some probability, instead of moving to a neighbor chosen uniformly at random, the walk moves to a vertex as determined by a memoryless controller whose aim is to reach a certain target vertex. The biased walk we analyze is a variant of *ϵ-biased walks* defined in [5]. In Section 5.1, we review the ϵ-biased walks of [5] and introduce and analyze our variant, which we call *inverse-degree-biased* walks. In Section 5.2, we use $1/d$-biased walks to derive an $O(n^{2-1/d})$ bound on the hitting time of the cobra walk on d-regular graphs, improving on the quadratic bound for standard random walks. In Section 5.3, we use inverse-degree-biased walks to derive an $O(n^{11/4})$ bound on the hitting time of the cobra walk on nonregular graphs, improving on the cubic bound for standard random walks.

5.1 Biased random walks

An *ϵ-biased walk* on G is defined as follows: in each step, with probability $1 - \epsilon$ one of the neighbors is selected uniformly at random, and the walk moves there; with probability ϵ, a controller gets to select which neighbor to move to. The controller can be probabilistic, but it is time independent. The primary motivation for biased random walks is to study how much can a controller increase the occupancy probabilities at a targeted subset S of the vertices in the stationary distribution. A central result of [5] is the following lower bound achieved by an optimal controller in an ϵ-biased walk.

THEOREM 11 (THEOREM 2 OF [5]). *Let $G = (V, E)$ be a connected graph, $S \subset V$, $v \in S$ and $x \in V$. Let $\Delta(x, v)$ be the length of the shortest path between vertices x and v in G and $\Delta(x, S) = \min_{v \in S} \Delta(x, v)$. Let $\beta = 1 - \epsilon$. There is an ϵ-bias strategy for which the stationary probability at S (i.e. the sum of the stationary probabilities of $v \in S$) is at least*

$$\frac{\sum_{v \in S} \delta(v)}{\sum_{v \in S} \delta(v) + \sum_{x \notin S} \beta^{\Delta(x,S)-1} \delta(x)}. \tag{1}$$

The bias in an ϵ-biased walk is constant (ϵ) at each vertex. To analyze the cobra walk, we define a biased walk in which the bias available to the controller at each vertex of the graph varies with the vertex. Specifically, we define the *inverse-degree-biased walk* with target x: if the walk is at x, then the next vertex visited is a neighbor of x chosen uniformly at random; if the walk is at vertex $v \neq x$, then with probability $1 - 1/\delta(v)$ one of the neighbors is selected uniformly at random and the walk moves there, and with probability $1/\delta(v)$ a controller gets to select which neighbor to move to. The goal of the controller is to minimize the hitting time to x.

Our main motivation for introducing inverse-degree-biased walks is the following dominance argument. For any pair of vertices u and v, let $H^*(u, v)$ denote the smallest hitting time to v achievable by an inverse-degree-biased walk with target v, starting at u. Recall that $H(u, v)$ denotes the hitting time at v for a cobra walk starting at u.

LEMMA 12. *For any pair of vertices u and v, $H(u, v) \leq H^*(u, v)$.*

PROOF. Our proof is by a coupling argument. Consider the cobra walk starting at u at time 0. For any vertex x and any time t, let $E^*_t(x)$ denote the event that x is active in the cobra walk at the start of round t. For any vertex x, any neighbor y of x, and any time t, let $P^*_t(x, y)$ be the probability that y is active at the start of round $t + 1$ conditioned on the event that x is active at the start of round t. That is,

$$P^*_t(x, y) = \Pr[E^*_{t+1}(y) \mid E^*_t(x)].$$

We next consider any inverse-degree-biased walk starting from u. Let $E_t(x)$ denote the event that the walk is at vertex x at the start of round t. Let $P'_t(x, y)$ be the probability that the inverse-degree-biased walk is at y at the start of round $t + 1$, conditioned on the event that the walk is at x at the start of round t. By the definition of the cobra walk and the inverse-degree-biased walk, we have the straightforward derivation.

$$\begin{aligned} P^*_t(x, y) &= 1 - \left(1 - \frac{1}{\delta(u)}\right)^2 \\ &\geq \frac{2}{\delta(u)} - \frac{1}{\delta(u)^2} \\ &= \frac{1}{\delta(u)} + \left(1 - \frac{1}{\delta(u)}\right) \frac{1}{\delta(u)} \\ &\geq P'_t(x, y). \end{aligned}$$

We now invoke a standard coupling argument to obtain that for any round t and any vertex x, the probability that x is active in the cobra walk at the start of round t is at least the probability that the inverse-degree-biased walk is at x at the start of round t. This also implies that the expected time for u to become active in the cobra walk is at most the expected time to hit x in the inverse-degree-biased walk. Setting $x = v$ and choosing the inverse-degree-biased walk to be one that minimizes the hitting time to v yields $H(u, v) \leq H^*(u, v)$. □

5.2 $O(n^{2-1/d})$ hitting time for d-regular graphs

THEOREM 13. *Let G be a finite, d-regular graph. Then the hitting time of a cobra walk on G is in $O(n^{2-\frac{1}{d}})$, for any constant d.*

PROOF. For the case of d-regular graphs, the inverse-degree-biased walk with target v is exactly a $1/d$-biased walk with target v. We invoke Theorem 11 for d-regular graphs with $\beta = 1 - 1/d$, and letting S equal $\{v\}$. We calculate a lower bound on stationary probability at v of the $1/d$-biased walk with target v, we need to determine the maximum value that $\sum_{x \neq v} \beta^{\Delta(x,v)-1}$ can take. We calculate this maximum value, over all graphs with maximum degree d, hence yielding an upper bound for d-regular graphs as well. Let n_i denote the number of vertices within i hops of v. Then, we have

$$\sum_{x \neq v} \beta^{\Delta(v,x)-1} = \sum_{i \geq 1} n_i \beta^{i-1}.$$

Since the degree is at most d, we have the following constraints on the n_is, given by the number of vertices within a certain number of hops from v.

$$\sum_{1 \leq i \leq j} n_i \leq \min \left\{ n - 1, d \left(1 + \sum_{1 \leq i < j} (d-1)^i \right) \right\}.$$

Since $\beta < 1$, and the total number of vertices is bounded by n, the sum $\sum_{i \geq 1} n_i \beta^{i-1}$ is maximized when

$$\begin{aligned} n_1 &= d \\ n_j &= d(d-1)^j \; 1 < j < L, \end{aligned}$$

where L is the smallest j such that $d\left(1 + \sum_{1 \leq i < j} (d-1)^i\right)$ is at least $n - 1$. Then, we have $n_L = n - 1 - \sum_{j<L} n_j$. With elementary algebra, we calculate the value of L as

$$L = \log_{(d-1)} \left[\frac{(n-1)}{d}(d-2) + 1 \right].$$

We now bound $\sum_{x \neq v} \beta^{\Delta(x,v)-1}$ as follows.

$$\begin{aligned} \sum_{x \neq v} \beta^{\Delta(v,x)-1} &= d\beta^0 + (d-1)d\beta^2 + d(d-1)^2\beta^2 \\ &\quad + \ldots + d(d-1)^{L-1}\beta^{L-1} \\ &= d \left(\sum_{i=0}^{L-1} ((d-1)\beta)^i \right) \\ &= d \left[\frac{((d-1)\beta)^L - 1}{(d-1)\beta - 1} \right] \\ &\leq C \left[((d-1)\beta)^L \right] \end{aligned}$$

where $C = d/((d-1)\beta - 1)$.

With some minimal algebra, we can see that:

$$(d-1)^{\log_{d-1}(n-1)(d-2)/d+1} = (n-1)(d-2)/d + 1 < n$$

and that

$$\begin{aligned}\beta^L &= (1-1/d)^{\log_{d-1}(n-1)(d-2)/(d-1)+1} \\ &\leq (d-1)^{(-1/d)\log_{d-1}((n-1)(d-1)/(d-2)+1)} \\ &= \left[\frac{(n-1)(d-2)+d}{d}\right]^{-1/d} < n^{-1/d}.\end{aligned}$$

Therefore, the stationary probability at v is at least $1/(1+n^{1-1/d})$, giving a return time to v of at most $1+n^{1-1/d}$. Using the tree-traversal argument and noting that it will take, in expectation, d returns to current vertex v to make one "progress" step in the traversal in expectation for a total cost of $dn^{1-1/d}$ per progress step. Thus, the total cost of coverage will be $2dCn^{2-1/d} = O(n^{2-1/d})$ for constant d. □

5.3 $O(n^{11/4})$ hitting time for general graphs

To bound $H(u,v)$ in a general graphs, let P be a shortest-hop path from u to v, given by the sequence of vertices $u = u_0, u_1, u_2, \ldots, u_{|P|} = v$. Then

$$H(u,v) \leq \sum_{i=0}^{|P|-1} H(u_i, u_{i+1}). \tag{2}$$

Let $R(x)$ denote the minimum return time to x among all inverse-degree-biased walks with target x. We have

$$R(x) \quad = \quad 1 + \sum_{y \in N(x)} \frac{H^*(y,x)}{\delta(x)}. \tag{3}$$

Furthermore, we have

$$\begin{aligned}H(u_i,u_{i+1}) &\leq 1 + \sum_{y\in N(u_i), y\neq u_{i+1}} H(y,u_{i+1})/\delta(u_i) \\ &\leq 1 + \sum_{y\in N(u_i), y\neq u_{i+1}} \frac{H(y,u_i)+H(u_i,u_{i+1})}{\delta(u_i)} \\ &\leq 1 + \sum_{y\in N(u_i), y\neq u_{i+1}} \frac{H^*(y,u_i)+H(u_i,u_{i+1})}{\delta(u_i)} \\ &\leq R(u_i) + \sum_{y\in N(u_i), y\neq u_{i+1}} H(u_i,u_{i+1})/\delta(u_i) \\ &= R(u_i) + \left(1 - \frac{1}{\delta(u_i)}\right) H(u_i,u_{i+1}).\end{aligned}$$

(The second step follows from the fact that the hitting time from y to u_{i+1} is at most the sum of the hitting times from y to u_i and from u_i to u_{i+1}. The third step follows from Lemma 12. The fourth step follows from Equation 3. The final equality holds since u_i has $\delta(u_i)$ neighbors.)

The above equation easily yields $H(u_i,u_{i+1}) \leq \delta(u_i)R(u_i)$, which combined with Equation 2 yields

$$H(u,v) \leq \sum_{i=0}^{|P|-1} H(u_i,u_{i+1}) \leq \sum_{i=0}^{|P|-1} \delta(u_i)R(u_i). \tag{4}$$

We prove the following extension of Theorem 11, applied to inverse-degree-biased walks. For the sake of completeness, we include the proof, which essentially mimics the proof of Theorem 2 in [5].

LEMMA 14. *Let $G = (V,E)$ be a connected graph and S be an arbitrary subset of V. For a given path P, let $\sigma(P)$ denote the product $\prod_{y\in P}(1-1/\delta(y))$. For $v \in S$ and $x \in V - S$, let $\widehat{\sigma}(x,v)$ denote the maximum, over all paths P from x to v, of $\sigma(P)$. Let $\widehat{\sigma}(x,S)$ denote the maximum, over all $v \in S$, of $\widehat{\sigma}(x,v)$. There is an inverse-degree-biased walk for which the stationary probability at S (i.e. the sum of the stationary probabilities of $v \in S$) is at least*

$$\frac{\sum_{v\in S}\delta(v)}{\sum_{v\in S}\delta(v) + \sum_{x\notin S}(\widehat{\sigma}(x,S)\delta(x))}.$$

PROOF. Define a probability distribution π^M: $\pi^M(v) = \gamma\delta(v)$ for $v \in S$ and $\pi^M(x) = \gamma\widehat{\sigma}(x,S)\delta(x)$ for $x \notin S$, where

$$\gamma = \frac{1}{\sum_{v\in S}\delta(v) + \sum_{x\notin S}(\widehat{\sigma}(x,S)\delta(x))}.$$

Let M be the transition probability matrix for which the stationary distribution given by Metropolis Theorem [5, Theorem 1] [22] is π^M. Define the following transition probability matrix: $P_{x,x} = 0$ for all $x \in V$, and for $y \neq x$, $P_{x,y} = M_{x,y}/(1-M_{x,x})$. We now argue that $M_{x,y} \geq (1-1/\delta(x))/\delta(x)$ for any neighbor y of x. By the Metropolis Theorem, $M_{x,y}$ is either $1/\delta(x)$ or $\pi^M(y)/(\delta(y)\pi^M(x))$. In the former case, the desired lower bound on $M_{x,y}$ is trivial. In the latter case, we have

$$M_{x,y} = \frac{\widehat{\sigma}(y,S)}{\delta(x)\widehat{\sigma}(x,S)} \geq \frac{1-1/\delta(x)}{\delta(x)},$$

where the last inequality follows from the fact that $\widehat{\sigma}(y,S) \geq (1-1/\delta(x))\widehat{\sigma}(x,S)$. We thus obtain $P_{x,y} \geq M_{x,y} \geq (1-1/\delta(x))/\delta(x)$ for any neighbor y of x. So P is a transition probability matrix of an inverse-degree-biased walk.

We next show that for each vertex in S, its stationary probability under P is at least its stationary probability under M. Let $H^M(x,v)$ and $H^P(x,v)$ denote the hitting time to v under the walks defined by P and M, respectively, with start vertex x. For each vertex $x \neq v$, we have

$$H^M(x,v) = 1 + M_{x,x}H^M(x,v) + \sum_{y\in N(x)} M_{x,y}H^M(y,v),$$

which yields

$$\begin{aligned}H^M(x,v) &= \frac{1}{1-M_{x,x}} + \sum_{y\in N(x)} \frac{M_{x,y}}{1-M_{x,x}} H^M(y,v) \\ &\geq 1 + \sum_{y\in N(x)} P_{x,y}H^M(y,v).\end{aligned}$$

By [5, Lemma 1], for every $x \in V$ and $v \in S$, $H^M(x,v) \geq H^P(x,v)$. If we denote the return time to vertex v in the walks under P and M by $R^P(v)$ and $R^M(v)$, respectively, we obtain that

$$\begin{aligned}R^M(v) &= 1 + \sum_{y\in N(v)} M_{v,y}H^M(y,v) \\ &= 1 + \sum_{y\in N(v)} P_{v,y}H^M(y,v) \\ &\geq 1 + \sum_{y\in N(v)} P_{v,y}H^P(y,v) \\ &= R^P(v).\end{aligned}$$

(The second step follows from the fact that for any v in S, and neighbor y of v, $M_{v,y} = P_{v,y} = 1/d(v)$.) We thus have

$$\pi^P(S) \geq \pi^M(S) = \frac{\sum_{v\in S}\delta(v)}{\sum_{v\in S}\delta(v) + \sum_{x\notin S}\delta(x)\widehat{\sigma}(x,S)}.$$

□

Corollary 15. *Let v be an arbitrary vertex of G. There exists a bias strategy for the inverse-degree-biased walk such that the return time to v in the walk is at most*

$$\frac{\delta(v) + \sum_{x \neq v} \widehat{\sigma}(x,v)\delta(x)}{\delta(v)}.$$

Proof. The result follows from Lemma 14, setting $S = \{v\}$ and noting that the return time to v equals the inverse of the stationary probability at v. □

For our analysis, it is more convenient to work with an upper bound on $\widehat{\sigma}(x,v)$ (for each x and v) as given the following lemma. Let $p(y,x)$ denote the weight of the shortest path from y to x in G, where we assign a weight of $1/\delta(z)$ for each vertex z.

Lemma 16. *For all vertices x and v, $\widehat{\sigma}(x,v)$ is at most $e^{-p(x,v)}$.*

Proof. Let P' denote a path between x and v such that $\widehat{\sigma}(x,v) = \sigma(P)$. Then, we have

$$\widehat{\sigma}(x,v) = \prod_{y \in P'} (1 - 1/\delta(y)) \le \prod_{y \in P'} e^{-1/\delta(y)} = e^{-p(x,v)}.$$

□

Lemma 17. *For start vertex u and target v, let P be a shortest-hop path from u to v, given by the sequence of vertices $u = u_0, u_1, u_2, \ldots, u_{|P|} = v$. Then, we have*

$$\sum_{i=0}^{|P|-1} \delta(u_i)R(u_i) = O(n^{11/4}).$$

Proof. By Lemma 12 and Corollary 15, we have

$$\begin{aligned}
\sum_{i=0}^{|P|-1} \delta(u_i)R(u_i) &\le \sum_{i=0}^{|P|-1} \delta(u_i)R(u_i) \\
&\le \sum_{i=0}^{|P|-1} \delta(u_i)\left(\frac{\delta(u_i) + \sum_{x \neq u} \widehat{\sigma}(x,u)\delta(x)}{\delta(u_i)}\right) \\
&\le \sum_{i=0}^{|P|-1} \left(\delta(u_i) + \sum_{x \neq u_i} \delta(x) e^{-p(x,u_i)}\right) \\
&\le 3n + \sum_{i=1}^{|P|-1} \sum_{x \neq u_i} \delta(x) e^{-p(x,u_i)} \\
&\le 3n + \sum_{x \in V} \left(\delta(x) \sum_{i=1}^{|P|-1} e^{-p(x,u_i)}\right).
\end{aligned}$$

(The second step follows from Corollary 15. The third step follows from Lemma 16. The fourth step follows from the elementary claim that the sum of the degrees of vertices along any shortest path in an n-vertex graph is at most $3n$; e.g., see [17, Proof of Theorem 2.1]. The final step is obtained by rearranging the summations.)

We next place an upper bound on $\sum_{x \in V} \left(\delta(x) \sum_{i=1}^{|P|-1} e^{-p(x,u_i)}\right)$. We claim that the number of vertices u_i in P to which any vertex x has paths of length at most L is at most $2L$. This is because two vertices a and b that are more than $2L$ hops away in P would have a path of $2L$ hops via x, a contradiction.

Consider a vertex u_i, for some i. Consider any path from x to u_i that minimizes $p(x,u_i)$. No vertex in the graph has edges to more than $\sqrt{n}$ vertices along this path. If it did, then we could short-cut and decrease the sum of the reciprocals of the degrees $1/\sqrt{n} < \sqrt{n}/(n-1)$. Therefore, any path that minimizes $p(x,u_i)$ has a total degree of at most $n^{3/2}$. If the length L of the path is less than $\sqrt{n}$, then $p(x,u_i) \ge 0$; otherwise, $p(x,u_i) \ge L^2/n^{3/2}$.

We now show that for any vertex x, $\sum_{i=1}^{|P|-1} e^{-p(x,u_i)}$ is $O(n^{3/4})$. We partition the u_i according to their hop-distance from x. Fix j and consider all vertices in P that are distance between 2^{j-1} and 2^j from x. By our claim above, there are at most 2^j such vertices. If $2^j \le \sqrt{n}$, then we use the trivial bound of 1; otherwise, we use the bound $2^{2j}/n^{3/2}$. So we have

$$\begin{aligned}
\sum_{u_i \in P} e^{-p(x,u_i)} &= \sum_{j:2^j \le \sqrt{n}} O(2^j) + \sum_{j:2^j > \sqrt{n}} \left(2^j * e^{-2^{2j}/n^{\ 3/2}}\right) \\
&= O(n^{1/2}) + O(n^{3/4}) = O(n^{3/4}).
\end{aligned}$$

Since $\delta(x)$ for each x is at most $n-1$ and the length of P is at most n, we get the desired bound of $O(n^{11/4})$. □

Theorem 18. *The hitting and cover times for a 2-cobra walk on a graph with n vertices are $O(n^{11/4})$ and $O(n^{11/4}\log n)$, respectively.*

Proof. For any start vertex u and target vertex v, let P be a shortest-hop path from u to v, given by the sequence of vertices $u = u_0, u_1, u_2, \ldots, u_{|P|} = v$. By Lemma 12 and Equation 4 we have

$$H(u,v) \le H^*(u,v) \le \sum_{i=0}^{|P|-1} \delta(u_i)R(u_i),$$

which, by Lemma 17, is $O(n^{11/4})$. The bound on the cover time follows from Matthews' Theorem, Theorem 1. □

6. CONCLUSION

We have derived several improved results for cobra walks. However, there remain many open problems, and corresponding gaps in our understanding. We have shown that for d-dimensional grids on $[0,n]^d$ the cover time is proportional to n, even for 2-cobra walks. The dependence on d, however, has not been determined, and more generally we would like to find a larger collection of graph types for which the cover time for 2-cobra walks is proportional to the diameter. Better results for general graphs are still open; we optimistically conjecture that for 2-cobra walks the worst case cover time on a graph with n vertices is only $O(n \log n)$. (The star graph shows that it can be $\Omega(n \log n)$.) Perhaps most importantly, we believe there should be better methods available for analyzing cobra walks. Our results utilize techniques based on parallelism and bias, but we do not believe they are as yet taking full advantage of the power of cobra walks.

7. REFERENCES

[1] Micah Adler, Eran Halperin, Richard M. Karp, and Vijay V. Vazirani. A stochastic process on the hypercube with applications to peer-to-peer networks. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 575–584, New York, NY, USA, 2003. ACM.

[2] Noga Alon, Chen Avin, Michal Koucky, Gady Kozma, Zvi Lotker, and Mark R. Tuttle. Many random walks are faster than one. In *Proceedings of the Twentieth Annual Symposium on Parallelism in Algorithms and Architectures*, SPAA '08, pages 119–128, New York, NY, USA, 2008. ACM.

[3] Siva R. Arthreya and Jan M Swart. Branching-coalescing particle systems. *Probability Theory and Related Fields*, 131(3):376–414, 2005.

[4] J. Augustine, G. Pandurangan, P. Robinson, S. Roche, and E. Upfal. Enabling robust and efficient distributed computation in dynamic peer-to-peer networks. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 350–369, Oct 2015.

[5] Yossi Azar, Andrei Z Broder, Anna R Karlin, Nathan Linial, and Steven Phillips. Biased random walks. In *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*, pages 1–9. ACM, 1992.

[6] Itai Benjamini and Sebastian Müller. On the trace of branching random walks. *arXiv preprint arXiv:1002.2781*, 2010.

[7] Petra Berenbrink, Colin Cooper, Robert Elsässer, Tomasz Radzik, and Thomas Sauerwald. Speeding up random walks with neighborhood exploration. In *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 1422–1435, Philadelphia, PA, USA, 2010. Society for Industrial and Applied Mathematics.

[8] Andrei Broder. Generating random spanning trees. In *Foundations of Computer Science, 1989., 30th Annual Symposium on*, pages 442–447, Oct 1989.

[9] Jen-Yeu Chen and Gopal Pandurangan. Almost-optimal gossip-based aggregate computation. *SIAM J. Comput.*, 41(3):455–483, 2012.

[10] F. Chung. Laplacians and the cheeger inequality for directed graphs. *Annals of Combinatorics*, 9:1–19, 2005.

[11] Colin Cooper, Robert Elsässer, Hirotaka Ono, and Tomasz Radzik. Coalescing random walks and voting on graphs. In *Proceedings of the 2012 ACM Symposium on Principles of Distributed Computing*, PODC '12, pages 47–56, New York, NY, USA, 2012. ACM.

[12] Nedialko B. Dimitrov and C. Greg Plaxton. Optimal cover time for a graph-based coupon collector process. In *Proceedings of the 32Nd International Conference on Automata, Languages and Programming*, ICALP'05, pages 702–716, Berlin, Heidelberg, 2005. Springer-Verlag.

[13] Chinmoy Dutta, Gopal Pandurangan, Rajmohan Rajaraman, and Scott Roche. Coalescing-branching random walks on graphs. *ACM Trans. Parallel Comput.*, 2(3):20:1–20:29, November 2015.

[14] Robert Elsasser and Thomas Sauerwald. Tight bounds for the cover time of multiple random walks. volume 412, pages 2623 – 2641, 2011. Selected Papers from 36th International Colloquium on Automata, Languages and Programming (ICALP 2009).

[15] U. Feige. A tight upper bound on the cover time for random walks on graphs. 1993.

[16] Uriel Feige. A tight lower bound on the cover time for random walks on graphs. *Random Struct. Algorithms*, 6(4):433–438, July 1995.

[17] Uriel Feige, David Peleg, Prabhakar Raghavan, and Eli Upfal. Randomized broadcast in networks. *Random Structures & Algorithms*, pages 447–460, 1990.

[18] Ayalvadi J. Ganesh, Laurent Massouli, and Donald F. Towsley. The effect of network topology on the spread of epidemics. In *INFOCOM*, pages 1455–1466. IEEE, 2005.

[19] Theodore E. Harris. *The theory of branching processes.* Die Grundlehren der Mathematischen Wissenschaften, Bd. 119. Springer-Verlag, Berlin, 1963.

[20] David A. Kessler. Epidemic size in the sis model of endemic infections. *Journal of Applied Probability*, 45(3):757–778, 09 2008.

[21] Neal Madras and Rinaldo Schinazi. Branching random walks on trees. *Stochastic Processes and their Applications*, 42(2):255 – 267, 1992.

[22] M. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and M. Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092, 1953.

[23] Michael Mitzenmacher, Rajmohan Rajaraman, and Scott Roche. Better bounds for coalescing-branching random walks, 2016. arXiv:1603.06109.

[24] Michael Mitzenmacher and Eli Upfal. *Probability and computing: Randomized algorithms and probabilistic analysis.* Cambridge University Press, 2005.

[25] Rongfeng Sun and Jan M Swart. The brownian net. *The Annals of Probability*, 36(3):1153–1208, 2008.

[26] Piet Van Mieghem. The n-intertwined sis epidemic network model. *Computing*, 93(2-4):147–169, December 2011.