Ant-Inspired Density Estimation via Random Walks

Cameron Musco[a,1], **Hsin-Hao Su**[a,1], **and Nancy Lynch**[a,1,2]

[a]Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139.

This manuscript was compiled on September 2, 2017

Many ant species employ distributed population density estimation in applications ranging from quorum sensing, to task allocation, to appraisal of enemy colony strength. It has been shown that ants estimate local population density by tracking encounter rates – the higher the density, the more often the ants bump into each other. We study distributed density estimation from a theoretical perspective. We prove that a group of anonymous agents randomly walking on a grid are able to estimate their density within a small multiplicative error in few steps by measuring their rates of encounter with other agents. Despite dependencies inherent in the fact that nearby agents may collide repeatedly (and, worse, cannot recognize when this happens), our bound nearly matches what would be required to estimate density by independently sampling grid locations. From a biological perspective, our work helps shed light on how ants and other social insects can obtain relatively accurate density estimates via encounter rates. From a technical perspective, our analysis provides new tools for understanding complex dependencies in the collision probabilities of multiple random walks. We bound the strength of these dependencies using *local mixing properties* of the underlying graph. Our results extend beyond the grid to more general graphs and we discuss applications to size estimation for social networks, density estimation for robot swarms, and random-walk-based sampling for sensor networks.

population density estimation | random walk sampling | network exploration | ant colony algorithms | biological distributed algorithms

The ability to sense local population density is an important tool used by many ant species. When a colony of *Temnothorax* ants must relocate to a new nest, scouts search for potential nest sites, assess their quality, and recruit other scouts to high quality locations. A high enough density of scouts at a potential new nest (a *quorum threshold*) triggers those ants to decide on the site and transport the rest of the colony there (1). When neighboring colonies of *Azteca* ants compete for territory, a high relative density of a colony's ants in a contested area will cause those ants to attack enemies in the area, while a low relative density will cause the colony to retreat (2). Varying densities of harvester ants successfully performing certain tasks such as foraging or brood care can trigger other ants to switch tasks, maintaining proper worker allocation in the colony (3, 4).

It has been shown that ants estimate density in a distributed manner, by measuring encounter rates (1, 5). As ants randomly walk around an area, if they bump into a larger number of other ants, this indicates a higher population density. By tracking encounters with specific types of ants, for example, successful foragers or enemies, ants can estimate more specific densities. This strategy allows each ant to obtain an accurate density estimate and requires very little communication – ants must simply detect when they collide and do not need to perform any higher level data aggregation.

Density Estimation on a Grid. We study distributed density estimation from a theoretical perspective. We model a colony of ants as a set of anonymous agents randomly placed on a two-dimensional grid. Computation proceeds in rounds, with each agent stepping in a random direction in each round. A *collision* occurs when two agents reach the same position in the same round and encounter rate is measured as the number of collisions an agent is involved in during a sequence of rounds divided by the number of rounds. Aside from collision detection, the agents have no other means of communication.

The intuition that encounter rate tracks density is clear. It is easy to show that, for a set of randomly walking agents, the *expected* encounter rate measured by each agent is exactly the density d – the number of agents divided by the grid size (see Lemma 2). However, it is unclear if encounter rate actually gives a good density estimate – that is, if the estimate is close to its expectation with high probability.

Consider agents positioned not on the grid, but on a complete graph. In each round, each agent steps to a uniformly random position and in expectation, the number of other agents it collides with in this step is d. Since each agent chooses its new location uniformly at random in each step, collisions are essentially *independent* between rounds. The agents are effectively taking independent Bernoulli samples with success probability d, and by a standard Chernoff bound, within $O\left(\frac{\log(1/\delta)}{d\epsilon^2}\right)$ rounds each obtains a $(1\pm\epsilon)$ multiplicative approximation to d with probability $1-\delta$.

On the grid graph, the picture is significantly more complex. If two agents are initially located near each other, they are

Significance Statement

Highly complex distributed algorithms are ubiquitous in nature: from the behavior of social insect colonies and bird flocks, to cellular differentiation in embryonic development, to neural information processing. In our research, we study biological computation theoretically, combining a *scientific perspective*, which seeks to better understand the systems being studied, with an *engineering perspective*, which takes inspiration from these systems to improve algorithm design. In this work, we focus on the problem of population density estimation in ant colonies, demonstrating that extremely simple algorithms, similar to those employed by ants, solve the problem with strong theoretical guarantees and have a number of interesting computational applications.

C.M., H.S., and N.L. designed and performed research and wrote the paper. C.M. and H.S. developed technical bounds.

The authors declare no conflict of interest.

[1] C.M., H.S., and N.L. contributed equally to this work.

[2] To whom correspondence should be addressed. E-mail: lynch@csail.mit.edu

more likely to collide via random walking. After a first collision, due to their proximity, they are likely to collide repeatedly in future rounds. Since the agents are anonymous, they cannot recognize repeat collisions, and even if they could, it is unclear that it would help. On average, compared to the complete graph, agents collide with fewer individuals and collide multiple times with those individuals that they do encounter, making encounter rates a less reliable estimate of population density.

Mathematically speaking, on a graph with a *fast mixing time* (6), like the complete graph, each agent's location is only weakly correlated with its previous locations. This ensures that collisions are also weakly correlated between rounds and encounter rate serves as a very accurate estimate of density. The grid graph on the other hand is *slow mixing* – agent positions and hence collisions are highly correlated between rounds, lowering the accuracy of encounter-rate-based estimation.

Results. Surprisingly, despite the high correlation between collisions, we show that encounter-rate-based density estimation on the grid is nearly as accurate as on the complete graph. After just $O\left(\frac{\log(1/\delta)\log\log(1/\delta)\log(1/d\epsilon)}{d\epsilon^2}\right)$ rounds, each agent's encounter rate is a $(1 \pm \epsilon)$ approximation to d with probability $1-\delta$ (Theorem 1). This matches performance on the complete graph up to a $\log\log(1/\delta)\log(1/d\epsilon)$ factor.

Technically, to bound accuracy on the grid, we obtain moment bounds on the number of times that two randomly walking agents collide over a set of rounds (Lemma 5). These bounds also apply to the number of equalizations (returns to origin) of a single walk. While *expected* random walk hitting times, return times, and collision rates are well studied for many graphs, including grid graphs (6–8), higher moment bounds and high probability results are much less common.

Our moment bounds show that, while the grid graph is slow mixing, it has strong *local mixing*. That is, random walks tend to spread quickly over a local area and not repeatedly cover the same nodes, making random-walk-based density estimation accurate. Significant work has focused on showing that random walk sampling is nearly as good as independent sampling for fast mixing expander graphs (9, 10). To the best of our knowledge, we are the first to extend this type of analysis to slowly mixing graphs, showing that strong local mixing is sufficient in many applications.

The key to the local mixing property of the grid is an upper bound on the probability that two random walks starting from the same position re-collide (or that a single random walk equalizes) after a certain number of steps (Lemma 3). We show that re-collision probability bounds imply collision moment bounds on general graphs, and apply this technique to extend our results to d-dimensional grids, regular expanders, and hypercubes. We discuss applications of our bounds to the task of estimating the size of a social network using random walks (11), obtaining improvements over prior work for networks with relatively slow global mixing times but strong local mixing. We also discuss connections to density estimation by robot swarms and random-walk-based sensor network sampling (12, 13).

1. Theoretical Model for Density Estimation

We consider a set of agents populating a two-dimensional torus with A nodes (dimensions $\sqrt{A} \times \sqrt{A}$). At each time step, each agent has an associated ordered pair *position*, which gives its coordinates on the torus. We assume that A is large – larger than the area agents traverse over the runtimes of our algorithms. We believe the torus model successfully captures the dynamics of density estimation on a surface, while avoiding complicating factors of boundary behavior on a finite grid.

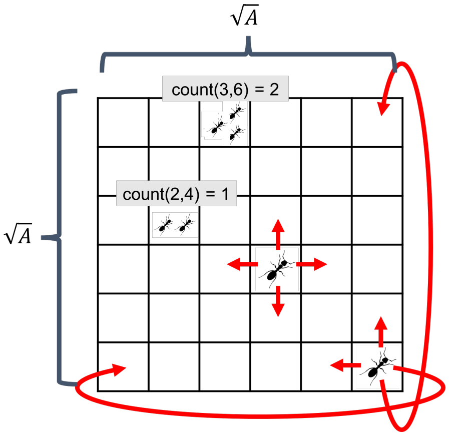

Fig. 1. A basic illustration of our computational model. Each agent (ant) may move to a random adjacent position on the two-dimensional torus in each round (illustrated by the red arrows). A collision occurs when two or more agents are located at the same position. The agents detect collisions through the $count(position)$ function which returns the *number of other agents* at their current position. In this illustration, $position$ is given as the (x, y) position with the bottom left corner corresponding to $(1, 1)$. However, the precise convention used is unimportant.

Initially each agent is placed independently at a uniform random node in the torus. Computation proceeds in discrete, synchronous rounds. Each agent updates its position with a step chosen uniformly at random from $\{(0,1),(0,-1),(1,0),(-1,0)\}$ in each round. Of course, in reality ants do not move via pure random walk – observed encounter rates seem to actually be lower than predicted by a pure random walk model (5, 14). However, we feel that our model sufficiently captures the highly random movement of ants while remaining tractable to analysis and applicable to ant-inspired random-walk-based algorithms (Section 4). Extending our work to more realistic models of ant movement would be an interesting next direction.

Aside from the ability to move in each round, agents can sense the number of agents other than themselves at their position at the *end of each round*, formally through the function *count*(*position*). We say that two agents *collide in round r* if they have the same position at the end of the round. Outside of collision counting, agents have no means of communication. They are anonymous (cannot uniquely identify each other) and execute identical density estimation routines. A basic illustration of our model is depicted in Figure 1.

The Density Estimation Problem. Let $(n+1)$ be the number of agents and define population density as $d \overset{\text{def}}{=} n/A$. Each agent's goal is to estimate d to $(1 \pm \epsilon)$ accuracy with probability at least $1-\delta$ for $\epsilon, \delta \in (0,1)$ – that is, to return an estimate $\tilde{d}$ with $\mathbb{P}\left[\tilde{d} \in [(1-\epsilon)d, (1+\epsilon)d]\right] \geq 1-\delta$. As a technicality, with $n+1$ agents we define $d = n/A$ instead of $d = (n+1)/A$ for convenience of calculation. In the natural case, when n is large, the distinction is unimportant.

Local vs. Global Density. The problem described above requires estimating the *global population density*. We assume that agents are initially distributed uniformly at random on the torus, which is critical for fast global density estimation – when agents are uniformly distributed, the local density in a small radius around their starting position reflects the global density with good probability. Of course, in nature, ants are not typically uniformly distributed in the nest or surrounding areas. Additionally, they are often interested in estimating *local population densities* – e.g., in a new nest site when house-hunting (1) or around a nest entrance when estimating the number of successful foragers for task allocation (3).

We view our work as a first step towards a theoretical understanding of density estimation and focus on the global density for simplicity. Removing our assumption of uniformly distributed agents and understanding local density estimation are important directions for future work.

2. Random-Walk-Based Density Estimation on the Two-Dimensional Torus

As discussed, the challenge in analyzing random-walk-based density estimation on the torus arises from correlations between collisions of nearby agents. If we do not restrict agents to random walking, and instead allow each agent to take an arbitrary step in each round, they can avoid collision correlations by splitting into 'stationary' and 'mobile' groups and counting collisions only between members of different groups. This allows them to essentially simulate independent sampling of grid locations to estimate density. This method is simple to analyze (see SI Appendix, Section S1), but it is not 'natural' in a biological sense or useful for the applications of Section 4. Further, independent sampling is unnecessary! Algorithm 1 describes a simple random-walk-based approach that gives a nearly matching bound.

Algorithm 1 Random-Walk-Based Density Estimation

```
Each agent independently executes:
  $c := 0$
  for $r = 1, \dots, t$ do
      $step := rand\{(0,1), (0,-1), (1,0), (-1,0)\}$
      $position := position + step$
      $c := c + count(position)$          ▷ Update collision count.
  return $\tilde{d} = \frac{c}{t}$
```

Our main theoretical result follows; its proof appears at the end of Section 2, after a number of preliminary lemmas. Throughout our analysis, we take the viewpoint of a single agent executing Algorithm 1.

Theorem 1 (Random Walk Sampling Accuracy Bound)**.** *After running for t rounds, assuming $t \leq A$, an agent executing Algorithm 1 returns $\tilde{d}$ such that, for any $\delta > 0$, with probability $\geq 1-\delta$, $\tilde{d} \in [(1-\epsilon)d, (1+\epsilon)d]$ for $\epsilon = \Theta\left(\sqrt{\frac{\log(1/\delta)\log(2t)}{td}}\right)$. In other words, for any $\epsilon, \delta \in (0,1)$ if $t = \Theta\left(\frac{\log(1/\delta)\log\log(1/\delta)\log(1/d\epsilon)}{d\epsilon^2}\right)$, $\tilde{d}$ is a $(1 \pm \epsilon)$ multiplicative estimate of d with probability $\geq 1-\delta$.*

Theorem 1 focuses on the density estimate of a single agent executing Algorithm 1. However, we note that if we set $\delta = \frac{\delta'}{n}$, then by a union bound, all n agents will have $\tilde{d} \in [(1-\epsilon)d, (1+\epsilon)d]$ with probability δ'. The required running time t will depend just logarithmically on δ' and n.

Correctness of Encounter Rate in Expectation. The first step in proving Theorem 1 is to show that the encounter rate $\tilde{d}$ is an unbiased estimator of d. This result in fact holds for any ants randomly walking on any regular graph.

Lemma 2 (Unbiased Estimator)**.** $\mathbb{E}\,\tilde{d} = d$.

Proof. We can decompose the collision bound c maintained by each agent in Algorithm 1 as the sum of collisions with different agents over different rounds. Specifically, give the n other agents arbitrary ids $1, 2, \dots, n$ and let $c_j(r)$ equal 1 the agent collides with agent j in round r, and 0 otherwise. By linearity of expectation, $\mathbb{E}\,c = \sum_{j=1}^{n}\sum_{r=1}^{t}\mathbb{E}\,c_j(r)$.

Since each agent is initially at a uniform random location and after any number of steps, is still at uniform random location, for all j, r, $\mathbb{E}\,c_j(r) = 1/A$. Thus, $\mathbb{E}\,c = nt/A = dt$ and $\mathbb{E}\,\tilde{d} = \mathbb{E}\,c/t = d$. □

We note that the torus is bipartite, and hence two agents initially located an odd number of steps away from each other will never meet via random walking. However, this fact does not change the expectation of $\tilde{d}$ computed above and in fact does not affect any of our following proofs.

With Lemma 2 in place, it remains to show that the encounter rate is close to its expectation with high probability and so provides a good estimate of density. In order to do this, we must bound the strength of correlations between collisions of nearby agents in successive rounds, which can decrease the accuracy of the encounter-rate-based estimate.

A Re-collision Probability Bound. The key to bounding collision correlations is bounding the probability of a re-collision between two agents in round $r+m$, assuming a collision in round r, which we do in this section.

Let $c_j = \sum_{r=1}^{t} c_j(r)$ be the total number of collisions with agent j. Due to the initial uniform distribution of the agents, the c_j's are all independent and identically distributed.

Each c_j is the sum of highly correlated random variables – due to the slow mixing of the grid, if two agents collide at round r, they are much more likely to collide in successive rounds. However, by bounding this re-collision probability, we are able to give strong moment bounds for the distribution of each c_j. We bound not only its variance, but all higher moments. This allows us to show that the average $\tilde{d} = \frac{1}{t}\sum_{j=1}^{n} c_j$ falls close to its expectation d with high probability, giving Theorem 1.

Lemma 3 (Re-collision Probability Bound)**.** *Consider two agents a_1 and a_2 randomly walking on a two-dimensional torus of dimensions $\sqrt{A} \times \sqrt{A}$. If a_1 and a_2 collide in round r, for any $m \geq 0$, the probability that a_1 and a_2 collide again in round $r+m$ is $\Theta\left(\frac{1}{m+1}\right) + O\left(\frac{1}{A}\right)$.*

Proof Sketch. The full proof of Lemma 3 is given in SI Appendix, Section S2. We sketch the main ideas here and illustrate them in Figure 2.

We first show that the probability that a_1 and a_2 re-collide in round $r+m$ is identical to the probability that a single $2m$-step random walk ends at its starting position.

The re-collision probability is the probability that a_1 and a_2 have identical displacements after taking m steps each. By symmetry of the random walk steps, this is equal to the probability that a_1's displacement vector is equal to the negative of a_2's. Furthermore, this is just the probability that their $2m$

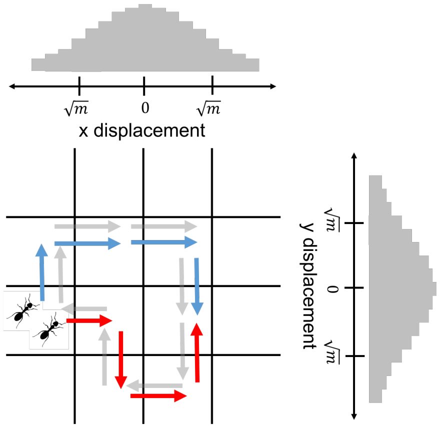

Fig. 2. A schematic of the proof of Lemma 3. We argue that the re-collision probability of two agents after m steps (shown in red and blue) is equivalent to the probability that a length $2m$ random walk (shown in grey) returns to its origin. We then argue that the random walk is likely to take roughly m steps in both the x and y directions and hence has zero displacement in each direction with probability $\Theta(1/\sqrt{m})$.

total random walk steps have 0 overall displacement, which is the probability that a $2m$-step random walk ends at its origin.

One idea might be to bound this 'equalization probability' using the global mixing time of the torus (6). After $\Theta(A \log A)$ steps, a random walk is nearly as likely to be at any node in the graph, including its origin. Thus the equalization probability is bounded by $O(\frac{1}{A})$ for $2m = \Omega(A \log A)$. Unfortunately, such a bound says nothing about this probability for small m.

Thus, we must take a different approach. We first assume for simplicity that the walk is on an infinite grid, and so there is no possibility of returning to its origin by 'wrapping around' the torus. We later show that this only affects the equalization probability by an $O\left(\frac{1}{A}\right)$ factor.

Considering a walk on the infinite grid, we condition on the walk taking roughly m steps in both the x and y directions, which occurs with high probability. We separately bound the probability of zero displacement in each direction.

It is well known that an m-step random walk on the line has roughly equal probability of ending at any point within radius $\Theta(\sqrt{m})$ of its origin. It thus has probability $\Theta\left(\frac{1}{\sqrt{m}}\right)$ of ending at its origin. Fixing the number of steps in each direction, the walk's x and y displacements are independent. So, we can multiply the probabilities for each direction, giving the final bound of $\Theta\left(\frac{1}{m+1}\right)$ (we write $m+1$ in the denominator instead of m so that the formula holds for $m = 0$.) □

Since it may be of independent interest, in Corollary 15 in SI Appendix, Section S3 we restate the result of Lemma 3 explicitly in terms of a bound on the probability that a single random walk returns to its origin (equalizes) after m steps.

Collision Moment Bound. With Lemma 3 in hand, we can prove our collision moment bound, which we use to show that the number of collisions an agent sees concentrates strongly around its expectation. We first show that any agent is likely to collide with many other agents during the execution of Algorithm 1, rather than repeatedly colliding with just a few other agents. That is, the probability that an agent collides at least once with any given other agent is not too low.

Lemma 4 (First Collision Probability)**.** *Assuming $t \le A$, for all $j \in [1, ..., n]$, $\mathbb{P}[c_j \ge 1] = \Theta\left(\frac{t}{A \log 2t}\right)$.*

Proof Sketch. By Lemma 3 and the assumption that $t \le A$, in t rounds, an agent expects to re-collide with any agent it encounters $\sum_{m=0}^{t-1} \Theta\left(\frac{1}{m+1}\right) = \Theta(\log 2t)$ times. By Lemma 2, an agent expects to be involved in $dt = nt/A$ total collisions. So accounting for re-collisions, it expects to collide with $\Theta\left(\frac{nt}{A \log 2t}\right)$ unique individuals. By symmetry, its collision probability with any single individual is thus $\Theta\left(\frac{t}{A \log 2t}\right)$. A formal proof is given in SI Appendix, Section S2. □

Lemma 4 used that by Lemma 3 an agent *expects* to collide $O(\log 2t)$ times with any other agent it encounters. We can in fact show that this bound is not just in expectation, but extends to the higher moments of the collision distribution.

Lemma 5 (Collision Moment Bound)**.** *For $j \in [1, ..., n]$, let $\bar{c}_j \overset{\text{def}}{=} c_j - \mathbb{E}\, c_j$ and assume $t \le A$. There is some fixed constant w such that for any integer $k \ge 2$,*

$$\mathbb{E}\left[\bar{c}_j^k\right] \le \frac{tw^k}{A} \cdot k! \log^{k-1}(2t).$$

When $k = 2$, Lemma 5 gives a bound on the *variance* of c_j, which can be used to show that c_j falls close to its mean with good probability. By bounding the k^{th} moment $\mathbb{E}[\bar{c}_j^k]$ for all k, we are able to show even stronger concentration results.

Proof Sketch. Very roughly, we separately consider the simple case when $c_j = 0$ and the case when $c_j \ge 1$, whose probability is bounded in Lemma 4. In the later case, we split c_j over rounds as $c_j = \sum_{r=1}^{t} c_j(r)$ and expand out:

$$\mathbb{E}[c_j^k] = \sum_{r_1=1}^{t} \sum_{r_2=1}^{t} ... \sum_{r_k=1}^{t} \mathbb{E}\left[c_j(r_1)c_j(r_2)...c_j(r_k)\right]. \qquad [1]$$

$\mathbb{E}\left[c_j(r_1)c_j(r_2)...c_j(r_k)\right]$ is just the probability that two agents collide in each of rounds $r_1, r_2, ..., r_k$. Assuming that $r_1 \le r_2 \le ... \le r_k$ and that there is a collision in round r_1, we can apply Lemma 3 to bound this probability $\le \frac{w^k}{(r_2-r_1+1)...(r_k-r_{k-1}+1)}$ for some constant w.

Obtaining the theorem requires combining this bound with Eq. (1) and applying a number of careful rearrangements. However, the bound on $\mathbb{E}\left[c_j(r_1)c_j(r_2)...c_j(r_k)\right]$ is the crux of the analysis. A full proof is in SI Appendix, Section S2. □

As with Lemma 3, the techniques used in Lemma 5 can be applied to bounding the moments of the number of equalizations of a single random walk. See Corollaries 16 and 17 in SI Appendix, Section S3.

Correctness of Encounter Rate With High Probability. Armed with Lemma 5 we can finally show that $\sum_{j=1}^{n} c_j$ concentrates strongly about its expectation. Since $\tilde{d} = \frac{1}{t}\sum_{j=1}^{n} c_j$, this is enough to prove the accuracy of encounter-rate-based density estimation (Algorithm 1). We first restate Lemma 5 using a standard 'Bernstein condition' on the sum $\sum_{j=1}^{n} c_j$.

Corollary 6 (Bernstein condition)**.** *Assuming $t \le A$:*

$$\mathbb{E}\left[\left(\sum_{j=1}^{n} c_j - \mathbb{E}\left[\sum_{j=1}^{n} c_j\right]\right)^k\right] \le \frac{1}{2} k! \sigma^2 b^{k-2}$$

for all $k \ge 2$ and some $b = \Theta(\log 2t)$ and $\sigma^2 = \Theta(td \log 2t)$.

Proof. By Lemma 5, there exists some constant w such that for $\sigma^2 = \frac{wt\log 2t}{A}$ and $b = w\log 2t$, $\bar{c}_j \overset{\text{def}}{=} c_j - \mathbb{E}\, c_j$ satisfies:

$$\mathbb{E}\left[\bar{c}_j^k\right] \le \frac{1}{2}k!\sigma^2 b^{k-2}.$$

Since each c_j is independent:

$$\mathbb{E}\left[\left(\sum_{j=1}^n c_j - \mathbb{E}\left[\sum_{j=1}^n c_j\right]\right)^k\right] = \mathbb{E}\left[\left(\sum_{j=1}^n \bar{c}_j\right)^k\right]$$

$$= \sum_{j=1}^n \mathbb{E}[\bar{c}_j^k] \le \frac{n\cdot k!\sigma^2 b^{k-2}}{2}.$$

The lemma follows after replacing σ^2 with $n\sigma^2 = \Theta(td\log 2t)$. □

We employ the following concentration bound for random variables satisfying such a Bernstein condition:

Lemma 7 (Proposition 2.3 of (15))**.** *Suppose that X satisfies $\mathbb{E}[(X - \mathbb{E}\, X)^k] \le \frac{1}{2}k!\sigma^2 b^{k-2}$ for all $k \ge 3$. Then for any $\Delta \ge 0$, $\mathbb{P}[|X - \mathbb{E}\, X| \ge \Delta] \le 2e^{-\frac{\Delta^2}{2(\sigma^2+b\Delta)}}$.*

We conclude this section by proving our main theorem on the accuracy of random-walk-based density estimation:

Proof of Theorem 1. In Algorithm 1, $\tilde{d}$ is set to $\frac{1}{t}\sum_{j=1}^n c_j$. So the probability that $\tilde{d}$ falls within an ϵ multiplicative factor of its mean is the same as the probability that $\sum_{j=1}^n c_j$ falls within an ϵ multiplicative factor of its mean, which is equal to $t\,\mathbb{E}\,\tilde{d} = td$ by Lemma 2. By Corollary 6 and Lemma 7:

$$\delta \overset{\text{def}}{=} \mathbb{P}\left[\left|\sum_{j=1}^n c_j - \mathbb{E}\left[\sum_{j=1}^n c_j\right]\right| \ge \epsilon\,\mathbb{E}\left[\sum_{j=1}^n c_j\right]\right]$$

$$= \mathbb{P}\left[\left|\sum_{j=1}^n c_j - td\right| \ge \epsilon td\right] \le 2e^{\Theta\left(-\frac{\epsilon^2t^2d^2}{2(td\log 2t+\epsilon td\log 2t)}\right)}.$$

Restricting $\epsilon \le 1$ and rearranging gives $\frac{\epsilon^2 td}{\log 2t} = \Theta\left(\log(1/\delta)\right)$ and so $\epsilon = \Theta\left(\sqrt{\frac{\log(1/\delta)\log 2t}{td}}\right)$, yielding the theorem. □

3. Extensions to Other Topologies

We now discuss extensions of our results to a broader set of graph topologies, demonstrating the generality of our local mixing analysis. We illustrate divergence between local and global mixing properties, which can have significant effects on random-walk-based algorithms. Full proofs for all results in this section are deferred to SI Appendix, Section S4.

From Re-collision Bounds to Accurate Density Estimation. Our proofs for the two-dimensional torus are largely independent of graph structure, using just a re-collision probability bound (Lemma 3) and the regularity (uniform node degrees) of the grid, so agents remain uniformly distributed on the nodes in each round (see for example, Lemma 2). Hence, extending our results to other regular graphs primarily involves obtaining re-collision probability bounds for these graphs.

We consider agents on a graph with A nodes that execute analogously to Algorithm 1, stepping to a random neighbor in each round. Again, we focus on the multi-agent case but similar bounds (resembling Corollaries 16 and 17) hold for a single random walk. We start with a lemma which gives density estimation accuracy in terms of re-collision probability. This is a direct generalization of our grid analysis.

Lemma 8 (Re-collision Probability to Density Estimation Accuracy)**.** *Consider a regular graph with A nodes such that, if two randomly walking agents a_1 and a_2 collide in round r, for any $0 \le m \le t$, the probability that they collide again in round $r+m$ is $\Theta(\beta(m))$ for some non-increasing function $\beta(m)$. Let $B(t) \overset{\text{def}}{=} \sum_{m=0}^t \beta(m)$. After running for $t \le A$ steps, Algorithm 1 returns $\tilde{d}$ such that, for any $\delta > 0$, with probability $\ge 1-\delta$, $\tilde{d} \in [(1-\epsilon)d, (1+\epsilon)d]$ for $\epsilon = O\left(\sqrt{\frac{\log(1/\delta)B(t)}{td}}\right)$.*

Note that in the special case of the two-dimensional torus, by Lemma 3, we can set $\beta(m) = 1/(m+1)$ and hence $B(t) = \Theta(\log 2t)$, recovering Theorem 1.

Density Estimation on k-Dimensional Tori. We first consider k-dimensional tori for general k. As k increases, local mixing becomes stronger, fewer re-collisions occur, and density estimation becomes easier. In fact, for constant $k \ge 3$, although the torus still mixes slowly, density estimation is as accurate as on the complete graph! Throughout this section we assume that k is a small constant and so hide multiplicative factors in $f(k)$ for any function f in our asymptotic notation. We subscript the notation with k to make this clear. For $k = 1$:

Lemma 9 (Re-collision Probability Bound – Ring)**.** *If two randomly walking agents a_1 and a_2 are located on a 1-dimensional torus (a ring) with A nodes, and collide in round r, for any $m \ge 0$, the probability that a_1 and a_2 collide again in round $r+m$ for $k \ge 1$ is $\Theta\left(\frac{1}{\sqrt{m+1}}\right) + O\left(\frac{1}{A}\right)$.*

Proof Sketch. This bound can be shown similarly to Lemma 3 (and in fact its proof is fully contained in the proof of Lemma 3.) A $2m$-step random walk on a line ends at its origin with probability $\Theta(1/\sqrt{m+1})$. On a ring with A nodes the slightly weaker bound of $\Theta\left(\frac{1}{\sqrt{m+1}}\right) + O\left(\frac{1}{A}\right)$ holds. □

For $m \le A$, the $O\left(\frac{1}{A}\right)$ term is absorbed into the $\Theta\left(\frac{1}{\sqrt{m+1}}\right)$ and one can show that $\sum_{m=0}^t 1/\sqrt{m+1} = \Theta(\sqrt{t})$. Plugging into Lemma 8, on a ring, random-walk-based density estimation gives $\epsilon = O\left(\sqrt{\frac{\log(1/\delta)\sqrt{t}}{td}}\right) = O\left(\sqrt{\frac{\log(1/\delta)}{\sqrt{t}d}}\right)$. Rearranging, $t = \Theta\left(\left(\frac{\log(1/\delta)}{\epsilon^2 d}\right)^2\right)$ rounds are necessary to obtain a $1 \pm \epsilon$ approximation with probability $\ge 1-\delta$ for any $\epsilon, \delta \in (0,1)$. Local mixing on the ring is much worse than on the torus. Hence, density estimation is much more difficult, requiring t to be quadratic rather than linear in $1/d$ and $1/\epsilon^2$.

We now cover $k \ge 3$. While global mixing time is on the order of $A^{2/k}$ (16), local mixing is so strong that our accuracy bounds nearly match those of independent sampling.

Lemma 10 (Re-collision Probability Bound – High-Dimensional Torus)**.** *If two randomly walking agents a_1 and a_2 are located on a k-dimensional torus with A nodes, and collide in round r, for any constant $k \ge 3$, $m \ge 0$, the probability that a_1 and a_2 collide in round $r+m$ is $\Theta_k\left(\frac{1}{(m+1)^{k/2}}\right) + O\left(\frac{1}{A}\right)$.*

Proof Sketch. The proof is similar to that of Lemma 3. To collide in round $r+m$, the agents must have identical displacements in each of the k dimensions after m steps. Since k is a small constant, with high probability the agents take $\Theta(m/k)$ steps in each dimension. After conditioning on the step counts, the k collisions are independent, each occurring with probability $\Theta\left(\frac{1}{\sqrt{m/k}}\right)$ via the argument of Lemma 3. The result follows by multiplying these k probabilities together, noting that k dependence is hidden in the asymptotic notation. □

To convert the above bound to a density estimation accuracy, we can use a slightly modified version of Lemma 8, which applies to the case when our collision probability is $O(\beta(m))$ but not neccesarily $\Theta(\beta(m))$. For $t \leq A$ and $k \geq 3$, $\sum_{m=0}^{t}\left(\frac{1}{(m+1)^{k/2}}+\frac{1}{A}\right) < 1+\sum_{m=0}^{\infty}\frac{1}{(m+1)^{k/2}} = O(1)$. So we can set $B(t) = O_k(1)$ and have $\epsilon = O_k\left(\frac{\sqrt{\log(1/\delta)}}{td}\right)$. Rearranging, we require $t = \Theta_k\left(\frac{\log(1/\delta)}{\epsilon^2 d}\right)$. This matches independent sampling up to constants and multiplicative factors in k.

Density Estimation on Regular Expanders. When a graph *does* mix well globally, it mixes well locally. An obvious example is the complete graph, on which random-walk-based and independent-sampling-based density estimation are equivalent. We extend this intuition to any regular expander. An expander is a graph whose random walk matrix has its second eigenvalue bounded away from 1, and so on which random walks mix quickly. Expanders are 'well-connected' graphs with many applications, including in the design of robust communication networks (17) and efficient sampling schemes (9).

Lemma 11 (Re-collision Probability Bound – Regular Expander)**.** *Let G be a k-regular expander with A nodes and adjacency matrix $\mathbf{M}$. Let $\mathbf{W} = \frac{1}{k}\cdot\mathbf{M}$ be its random walk matrix, with eigenvalues $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_A$. Let $\lambda = \max\{|\lambda_2|, |\lambda_A|\} < 1$. If two randomly walking agents a_1 and a_2 collide in round r, for any $m \geq 0$, the probability that they collide again in round $r+m$ is at most $\lambda^m + 2/A$.*

Proof Sketch. The bound follows from noting that the stable distribution on a regular expander is uniform, and the location distribution of any agent after m steps converges exponentially quickly to this distribution, with rate λ. □

Again, we bound density estimation accuracy via a modification of Lemma 8, which applies when we have collision probability $O(\beta(m))$ but not necessarily $\Theta(\beta(m))$. This modified lemma gives a $B(t)^2$ dependence. $B(t) = \sum_{m=0}^{t}\beta(m) \leq \frac{1}{1-\lambda}+2t/A$. Assuming $t = O(A)$, $\epsilon = O\left(\sqrt{\frac{\log(1/\delta)}{td(1-\lambda)^2}}\right)$. Rearranging, $t = \Theta\left(\frac{\log(1/\delta)}{\epsilon^2 d(1-\lambda)^2}\right)$, matching independent sampling up to a factor of $O(1/(1-\lambda)^2)$.

Density Estimation k-Dimensional Hypercubes. Finally, we give bounds for a k-dimensional hypercube. Such a graph has $A = 2^k$ vertices mapped to the elements of $\{\pm 1\}^k$, with an edge between any two vertices that differ by hamming distance 1. The hypercube is relatively fast mixing. Its adjacency matrix eigenvalues are $[-k, -k+2, ..., k-2, k]$. Since it is bipartite, we can ignore the negative eigenvalues: to return to its origin, a random walk must take an even number of steps, so we need only need to consider the squared walk matrix $\mathbf{W}^2$, which has all positive eigenvaules. Applying Lemma 11 with $\lambda = \Theta(1-2/k) = \Theta(1-1/\log A)$, gives $t = \Theta\left(\frac{\log(1/\delta)\log^2(A)}{\epsilon^2 d}\right)$. However, it is possible to remove the dependence on A via a more refined analysis – while the global mixing time of the graph increases as A grows, local mixing becomes stronger!

Lemma 12 (Re-collision Probability Bound – k-Dimensional Hypercube)**.** *If two randomly walking agents a_1 and a_2 are located on a k-dimensional hypercube with $A = 2^k$ vertices and collide in round r, for any $m \geq 0$, the probability that a_1 and a_2 collide in round $r+m$ is $O\left((7/10)^m + \frac{1}{\sqrt{A}}\right)$.*

Converting to a density estimation bound, we have $B(t) = \sum_{m=0}^{t}\beta(m) \leq \frac{10}{3} + t/\sqrt{A}$. If we assume $t = O(\sqrt{A})$, this gives $\epsilon = O\left(\sqrt{\frac{\log(1/\delta)}{td}}\right)$ and so $t = \Theta\left(\frac{\log(1/\delta)}{\epsilon^2 d}\right)$, matching independent sampling.

4. Applications

We conclude by discussing algorithmic applications of our ant-inspired density estimation algorithm (Algorithm 1), variations on this algorithm, and the analysis techniques we develop.

Social Network Size Estimation. Random-walk-based density estimation is closely related to work on estimating the size of social networks and other massive graphs using random walks (11, 18–20). In these applications, one does not have access to the full graph (so cannot exactly count the nodes), but can simulate random walks by following links between nodes (21, 22). One approach is to run a single random walk and count repeat node visits (18, 19). Alternatively, (11) proposes running multiple random walks and counting their collisions, which gives an estimate of the walk's density. Since the number of walks is known, this yields an estimate for network size.

This approach can be significantly more efficient since the dominant cost is typically in link queries to the network. With multiple, shorter random walks, this cost can be trivially distributed to multiple servers simulating walks independently. Visit information can then be aggregated and the collision count can be computed in a centralized manner.

Random-Walk-Based Algorithm for Network Size Estimation. Consider an undirected, connected, non-bipartite graph $G = (V, E)$. Let S be the set of vertices of G that are 'known'. Initially, $S = \{v\}$ where v is a seed vertex. We can access G by looking up the neighborhood $\Gamma(v_i)$ of any vertex $v_i \in S$ and adding $\Gamma(v_i)$ to S.

To compute the network size $|V|$, we could scan S, looking up the neighbors of each vertex and adding them to the set. Repeating this process until no new nodes are added ensures that $S = V$ and we know the network size. However, this method requires $|V|$ neighborhood queries. The goal is to use significantly fewer queries using random-walk-based sampling.

A number of challenges are introduced by this application. While we can simulate many random walks on G, we can no longer assume these random walks start at randomly chosen nodes, as we do not have the ability to uniformly sample nodes from the network. Instead, we must allow the random walks to run for a burn-in phase of length proportional to the mixing time of G. After this phase, the walks are distributed approximately according to the stable distribution of G.

Further, in general G is not regular. In the stable distribution, a random walk is located at a vertex with probability proportional to its degree. Hence, collisions tend to occur more at higher degree vertices. To correct for this bias, we count a collision at vertex v_i with weight $1/\deg(v_i)$.

Our results depend on a natural generalization of re-collision probability. For any i, j, let $p(v_i, v_j, m)$ be the probability that an m step random walk starting at v_i ends at v_j. Define:

$$\beta(m) \overset{\text{def}}{=} \frac{\max_{i,j} p(v_i, v_j, m)}{\deg(v_j)}.$$

Intuitively, $\beta(m)$ is the maximum m step collision probability, weighted by degree since higher degree vertices are visited more in the stable distribution. Let $B(t) = \sum_{m=1}^{t} \beta(m)$. Note that this weighted $B(t)$ is trivially upper bounded by the unweighted measure used in Lemma 8.

For simplicity, we initially ignore burn-in and assume that our walks start distributed exactly by the stable distribution of G. A walk starts at vertex v_i with probability $p_i \overset{\text{def}}{=} \frac{\deg(v_i)}{\sum_i \deg(v_i)} = \frac{\deg(v_i)}{2|E|}$ and initial locations are independent. We also assume knowledge of the average degree $\overline{\deg} = 2|E|/|V|$. See SI Appendix, Section S5 for a rigorous analysis of burn-in and average degree estimation.

Algorithm 2 Random-Walk-Based Network Size Estimation

```
input: step count $t$, average degree $\overline{\deg}$, $n$ random starting locations $[w_1, ..., w_n]$ distributed independently according to the network's stable distribution
  $[c_1, ..., c_n] := [0, 0, ..., 0]$
  for $r = 1, ..., t$ do
      $\forall j$, set $w_j := randomElement(\Gamma(w_j))$ ▷ $\Gamma(w_j)$ denotes the neighborhood of $w_j$.
      $\forall j$, set $c_j := c_j + \frac{count(w_j)}{\deg(w_j)}$ ▷ $count(w_j)$ returns the number of other walkers currently at $w_j$.
  $C := \frac{\overline{\deg} \sum c_j}{n(n-1)t}$
  return $\tilde{A} = 1/C$
```

Note that there are many ways to implement the $count(\cdot)$ function used in Algorithm 2. One possibility is to simulate the random walks in parallel, recording their paths, and then to perform centralized post-processing to count collisions. As queries to the network are considered to dominate time cost, this collision counting step is relatively inexpensive.

We prove the following theorem in SI Appendix Section S5:

Theorem 13. *If Algorithm 2 is run using n random walks for t steps, as long as $n^2 t = \Theta\left(\frac{B(t)\overline{\deg}+1}{\epsilon^2\delta} \cdot |V|\right)$, then with probability at least $1-\delta$, it returns $\tilde{A} \in [(1-\epsilon)|V|, (1+\epsilon)|V|]$.*

Proof Sketch. The proof is similar to that of Theorem 1. It is not hard to see that due to our reweighting of each collision by $1/\deg(w_j)$, $\mathbb{E}C = 1/|V|$. The challenge is showing that C concentrates around its expectation and hence $\tilde{A} = 1/C$ is close to $|V|$. Due to the complicating factors of non-uniform degree, we are unable to compute a general moment bound for each c_j as done in Lemma 5. However, we can give a variance bound on C, and bound its deviation via Chebyshev's inequality. This gives a worse dependence on the failure probability: $1/\delta$ instead of $\log(1/\delta)$. We note that this can be improved by running the algorithm $\log(1/\delta)$ times, each with success probability 1/3 and taking the median of the results. □

Overall Runtime and Comparision to Previous Work. Let M denote the burn-in time required before running Algorithm 2 (see SI Appendix Section S5 for details). In order to obtain a $(1 \pm \epsilon)$ estimate of network size with probability $1-\delta$ we must run n random walks for $M+t$ steps, making $n(M+t)$ link queries, where by Theorem 13, and our analysis of average degree estimation in SI Appendix Section S5 we have:

$$n = \Theta\left(\max\left\{\frac{\overline{\deg}}{\deg_{\min}\epsilon^2\delta}, \sqrt{\frac{|V| \cdot (B(t)\overline{\deg}+1)}{t \cdot \epsilon^2\delta}}\right\}\right). \quad [2]$$

Typically, the second term dominates since $\overline{\deg} << |V|$. Hence, by increasing t, we are able to use fewer random walks, significantly decreasing the number of link queries if M is large.

(11) uses a different approach, halting random walks and counting collisions immediately after burn-in. For reasonable node degrees they require $n = \Theta\left(\frac{|V| \cdot \overline{\deg}}{\epsilon^2\delta \cdot \sqrt{\sum \deg(v_i)^2}}\right)$. Assuming that $\sqrt{\sum \deg(v_i)^2} < n$, and setting $t = 1$, this is somewhat smaller than our bound as $\sum \deg(v_i)^2 \geq |V| \cdot \overline{\deg}$. However, Eq. (5) gives an important tradeoff – by increasing t we can increase the number of steps in our random walks, decreasing the total number of walks.

As an illustrative example, consider a k-dimensional torus graph for $k \geq 3$ (for $k = 2$ mixing time is $\Theta(|V|)$ so we might as well census the full graph). The mixing time required for Algorithm 2 (see SI Appendix Section S5 for details) is $M = \Theta(\log(|V|/\delta)|V|^{2/k})$. All nodes have degree $2k$, and using the bounds above, to obtain a $(1 \pm \epsilon)$ estimate of $|V|$, the algorithm of (11) requires $M \cdot n = \Theta\left(\frac{\log(|V|/\delta)}{\epsilon\sqrt{d}} \cdot |V|^{2/k+1/2}\right)$ link queries to obtain a size estimate. In contrast, assuming $|V|$ is large, we require $n = \Theta\left(\sqrt{\frac{|V|}{t \cdot \epsilon^2\delta}}\right)$ since by Lemma 10, $B(t) = O(1/k)$ and $\overline{\deg} = \deg_{\min} = k$. If we set $t = \Theta(M)$, the total number of link queries needed is $n(M+t) = O\left(\frac{\sqrt{\log(|V|/\delta)}}{\epsilon\sqrt{d}} \cdot |V|^{(k+1)/2k}\right)$. This beats (11) by improving dependence on $|V|$ and the logarithmic burn-in term. Ignoring error dependences, if $k = 3$, (11) requires $\Theta(n^{7/6})$ queries which is more expensive than fully censusing the graph. We require $O(n^{2/3})$ queries, which is sublinear in the graph size.

We leave open comparing our bounds with those of (11) on more natural classes of graphs. It would be interesting to determine typical values of $B(t)$ in real work networks or popular graph models, such as preferential attachment models and others with power-law degree distributions.

Distributed Density Estimation by Robot Swarms. Algorithm 1 can be directly applied as a simple and robust density estimation algorithm for robot swarms moving on a two-dimensional plane modeled as a grid. Additionally, the algorithm can be used to estimate the frequency of certain properties within the swarm. Let d be the overall population density and d_P be the density of agents with some property P. Let $f_P = d_P/d$ be the relative frequency of P.

Assuming that agents with property P are distributed uniformly in population and that agents can detect this property (through direct communication or some other signal), then they can separately track encounters with these agents. They can compute an estimate $\tilde{d}$ of d and $\tilde{d}_P$ of d_P. By Theorem 1, after running for $t = \Theta\left(\frac{\log(1/\delta)\log\log(1/\delta)\log(1/d\epsilon)}{d_P\epsilon^2}\right)$ steps, with probability $1-2\delta$, $\tilde{d}_P/\tilde{d} \in \left[\left(\frac{1-\epsilon}{1+\epsilon}\right)f_P, \left(\frac{1+\epsilon}{1-\epsilon}\right)f_P\right] = [(1-O(\epsilon))f_P, (1+O(\epsilon))f_P]$ for small ϵ.

In an ant colony, properties may include whether or not an ant has recently completed a successful foraging trip (3), or if an ant is a nestmate or enemy (2). In a robotics setting, properties may include whether a robot is part of a certain task group, whether it has completed a certain task, or whether it has detected a certain event or environmental property.

Random-Walk-Based Sensor Network Sampling. Finally, we believe our moment bounds for a single random walk (Corollaries 16 and 17) can be applied to random-walk-based distributed algorithms for sensor network sampling. We leave obtaining rigorous bounds in this domain to future work.

Random-walk-based sensor network sampling (12, 13) is a technique in which a query message (a 'token') is initially sent by a base station to some sensor. The token is relayed randomly between sensors, which are connected via a grid network, and its value is updated appropriately at each step to give an answer to the query. This scheme is robust and efficient - it easily adapts to node failures and does not require setting up or storing spanning tree communication structures.

Random-walk-based sampling could be used, for example, to estimate the percentage of sensors that have recorded a specific condition, or the average value of some measurement at each sensor. However, as in density estimation, unless an effort is made to record which sensors have been previously visited, additional error is added due to repeat visits. Recording previous visits introduces computational burden – either the token message size must increase or nodes themselves must remember which tokens they have seen. We are hopeful that our moment bounds can be used to show that this is unnecessary – due to strong local mixing, the number of repeat sensor visits will be low, and the performance reduction limited.

We remark that estimating the percentage of sensors in a network or the density of robots in a swarm with a property that is uniformly distributed is a special case of a more general *data aggregation* problem: each agent or sensor holds a value v_i drawn independently from some distribution $\mathcal{D}$. The goal is to estimate some statistic of $\mathcal{D}$, such as its expectation. In the case of density estimation, v_i is simply an indicator random variable which is 1 with probability d and 0 otherwise. Extending our results to more general data aggregation problems and showing that random walk sampling matches independent sampling in some cases is an interesting future direction.

5. Discussion and Future Work

We have presented a theoretical analysis of random-walk-based density estimation by agents moving synchronously on a two-dimensional torus graph. We have also presented applications of our techniques to density estimation on other simple graph topologies and to the problems of social network size estimation and density estimation on robot swarms.

Aside from using our bounds to study sensor network sampling and giving improved theoretical and empirical understanding of our social network size estimation algorithm, our work leaves open a number of open questions related to modeling random-walk-based density estimation in ant colonies.

We feel that our simple computational model well reflects the behavior of ants estimating density via collision rates while moving around a two-dimensional surface. However, extending our results to more realistic models, e.g., with continuous movement along a surface which is either bounded or extends out indefinitely, is an interesting future direction.

As discussed, understanding how close actual ant movements are to random walks, and how non-random behavior influences density estimation via collision detection is also important. In conjunction with this issue, removing our uniform density assumption and understanding how ants may estimate local population densities which may vary throughout the nest or surrounding area is an important direction.

Finally, we note that the accuracy bound of Theorem 1 depends on the density d. In many applications, such as in quorum sensing, ants only need to detect when d is above some fixed threshold. In this case, better bounds, where t can be determined independently of the density, may be possible.

ACKNOWLEDGMENTS. We thank Amartya Shankha Biswas, Christopher Musco, and Mira Radeva for useful discussions. We also thank Yehuda Afek, Ziv Bar-Joseph, and Amos Korman for many helpful comments and suggestions. Research was supported by NSF Grants BIO-1455983 and CCF-1461559, NSF CSoI grant CCF-0939370, and AFOSR grant FA9550-13-1-0042. Cameron Musco is partially supported by an NSF graduate student fellowship.

1. Pratt SC (2005) Quorum sensing by encounter rates in the ant *Temnothorax albipennis*. *Behavioral Ecology* 16(2):488–496.
2. Adams ES (1990) Boundary disputes in the territorial ant *Azteca trigona*: effects of asymmetries in colony size. *Animal Behaviour* 39(2):321–328.
3. Gordon DM (1999) Interaction patterns and task allocation in ant colonies in *Information Processing in Social Insects.* (Springer), pp. 51–67.
4. Schafer RJ, Holmes S, Gordon DM (2006) Forager activation and food availability in harvester ants. *Animal Behaviour* 71(4):815–822.
5. Gordon DM, Paul RE, Thorpe K (1993) What is the function of encounter patterns in ant colonies? *Animal Behaviour* 45(6):1083–1100.
6. Lovász L (1993) Random walks on graphs: A survey. *Combinatorics, Paul Erdos is Eighty* 2(1):1–46.
7. Elsässer R, Sauerwald T (2009) Tight bounds for the cover time of multiple random walks in *Automata, Languages and Programming.* (Springer), pp. 415–426.
8. Kanade V, Mallmann-Trenn F, Sauerwald T (2016) On coalescence time in graphs–when is coalescing as fast as meeting? *arXiv preprint arXiv:1611.02460.*
9. Gillman D (1998) A Chernoff bound for random walks on expander graphs. *SIAM Journal on Computing* 27(4):1203–1220.
10. Chung KM, Lam H, Liu Z, Mitzenmacher M (2012) Chernoff-Hoeffding bounds for Markov chains: Generalized and simplified. *arXiv preprint arXiv:1201.0559.*
11. Katzir L, Liberty E, Somekh O, Cosma IA (2014) Estimating sizes of social networks via biased sampling. *Internet Mathematics* 10(3-4):335–359.
12. Avin C, Brito C (2004) Efficient and robust query processing in dynamic environments using random walk techniques in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks.* (ACM), pp. 277–286.
13. Lima L, Barros J (2007) Random walks on sensor networks in *5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops, 2007.* (IEEE), pp. 1–5.
14. Nicolis SC, Theraulaz G, Deneubourg JL (2005) The effect of aggregates on interaction rate in ant colonies. *Animal Behaviour* 69(3):535–540.
15. Wainwright MJ (2015) High-dimensional statistics: A non-asymptotic viewpoint, draft (http://www.stat.berkeley.edu/~mjwain/stat210b/Chap2_TailBounds_Jan22_2015.pdf).
16. Aldous D, Fill J (2002) Reversible Markov chains and random walks on graphs.
17. Bassalygo L, Pinsker M (1973) The complexity of an optimal non-blocking commutation scheme without reorganization. *Problemy Peredaci Informacii* 9(1):84–87.
18. Kurant M, Butts CT, Markopoulou A (2012) Graph size estimation. *arXiv:1210.0460.*
19. Lu J, Li D (2012) Sampling online social networks by random walk in *Proceedings of the First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research.* (ACM), pp. 33–40.
20. Lu J, Wang H (2014) Variance reduction in large graph sampling. *Information Processing & Management* 50(3):476–491.
21. Mislove A, Marcon M, Gummadi KP, Druschel P, Bhattacharjee B (2007) Measurement and analysis of online social networks in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement.* (ACM), pp. 29–42.
22. Gjoka M, Kurant M, Butts CT, Markopoulou A (2009) A walk in Facebook: Uniform sampling of users in online social networks. *arXiv preprint arXiv:0906.0060.*

Supporting Information Appendix

S1: Density Estimation via Simulation of Independent Sampling

Here we show that, if agents are not restricted to random walking, but can instead take arbitrary steps in each round, they can avoid collision correlations by splitting into 'stationary' and 'mobile' groups and counting collisions only between members of different groups. This allows them to essentially simulate independent sampling of grid locations to estimate density. This algorithm is not 'natural' in a biological sense, however it is easy to analyze, and demonstrates the feasibility of density estimation by anonymous agents on the grid. We give pseudocode in Algorithm 3. Recall that *position* is an ordered pair denoting an agent's (x, y) coordinates on the torus graph, and *count*(*position*) returns the number of other agents at the current position.

Algorithm 3 Independent-Sampling-Based Density Estimation

```
input: runtime $t$
  Set $c := 0$ and with probability 1/2, $state := walking$, else $state := stationary$.
  for $r = 1, ..., t$ do
    if $state := walking$ then
      $position := position + (0, 1)$                ▷ Deterministic walk step.
    $c := c + count(position)$                      ▷ Update collision count.
  return $\tilde{d} = \frac{2c}{t}$
```

Theorem 14 (Independent Sampling Accuracy Bound)**.** *After running for t rounds, assuming $t < \sqrt{A}$ and $d \leq 1$, Algorithm 3 returns $\tilde{d}$ such that, for any $\delta > 0$, with probability $\geq 1 - \delta$, $\tilde{d} \in [(1-\epsilon)d, (1+\epsilon)d]$ for $\epsilon = \Theta\left(\sqrt{\frac{\log(1/\delta)}{td}}\right)$. In other words, for any $\epsilon, \delta \in (0, 1)$ if $t = \Theta\left(\frac{\log(1/\delta)}{d\epsilon^2}\right)$, $\tilde{d}$ is a $(1 \pm \epsilon)$ multiplicative estimate of d with probability $\geq 1 - \delta$.*

Proof. Our analysis is from the perspective of an agent with $state = walking$. By symmetry, the distribution of $\tilde{d}$ is identical for walking and stationary agents, so considering this case is sufficient.

Initially, assume that no two walking agents start in the same location. Given this assumption, we know that a walking agent *never collides with another walking agent* – by assumption they all start in different positions and update these positions identically in each round. In the written implementation agents always step up, however any fixed pattern (for example, a spiral) suffices.

In t steps, a walking agent visits t unique squares. Each of the n other agents is located in this set of squares *and* stationary with probability $\frac{t}{2A}$. Further, each of these events is entirely independent from the rest, as the agents are positioned and choose their state independently. So, for a walking agent, c is just a sample of n independent random coin flips, each with success probability $\frac{t}{2A}$. Clearly, $\mathbb{E}\, c = n \cdot \frac{t}{2A} = \frac{td}{2}$ so $\mathbb{E}\, \tilde{d} = \mathbb{E}\, \frac{2c}{t} = d$. Further, by a Chernoff bound, for any $\epsilon \in (0, 1)$, the probability that $\tilde{d}$ is not a $(1 \pm \epsilon)$ multiplicative estimate of d is:

$$\delta = \mathbb{P}\left[|\tilde{d} - d| \geq \epsilon d\right] = \mathbb{P}\left[|c - \mathbb{E}\, c| \geq \epsilon\, \mathbb{E}\, c\right] \leq 2e^{-\epsilon^2\, \mathbb{E}\, c/3} \leq 2e^{-\epsilon^2 td/6}.$$

This gives $\log(1/\delta) \geq \epsilon^2 td/6$ so we can set $\epsilon = \Theta\left(\sqrt{\frac{\log(1/\delta)}{td}}\right)$, yielding the result.

We now remove the assumption that no two walking agents start in the same location. We slightly modify the algorithm – each agent sets $c := c \pmod{t}$ before returning $\tilde{d} = \frac{2c}{t}$. If an agent starts alone and is involved in $< t$ collisions, this operation has no effect – the above bound holds.

If a walking agent is involved in $< t$ 'true collisions' but starts in the same position as $w \geq 1$ other walking agents, the agents move in lockstep throughout the algorithm and are involved in $w \cdot t$ 'spurious collisions' (w in each round). Setting $c := c \pmod{t}$ exactly corrects for these spurious collisions and since c now includes only collisions with stationary agents, the bound above holds.

Finally, if an agent is involved in $\geq t$ true collisions, this modification cannot worsen their estimate. If $c \geq t$ and the agent does not set $c := c \pmod{t}$, they compute $\tilde{d} \geq \frac{2t}{t} \geq 2$. For $\epsilon < 1$, the agent fails since by assumption $d \leq 1$. So setting $c := c \pmod{t}$ can only increase success probability. □

S2: Complete Proofs for Random-Walk-Based Density Estimation

We first prove our bound on the probability that two agents located in the same position at round r re-collide in round m.

Lemma 3 (Re-collision Probability Bound)**.** *Consider two agents a_1 and a_2 randomly walking on a two-dimensional torus of dimensions $\sqrt{A} \times \sqrt{A}$. If a_1 and a_2 collide in round r, for any $m \geq 0$, the probability that a_1 and a_2 collide again in round $r+m$ is $\Theta\left(\frac{1}{m+1}\right) + O\left(\frac{1}{A}\right)$.*

Proof. From round r to round $r + m$, a_1 and a_2 take $2m$ random steps in total. Let M_x be the total number of steps they take in the x direction and M_y be the total number in the y direction. $M_x + M_y = 2m$.

We start by computing the probability that the agents collide in round $r + m$ conditioned on the values of M_x and M_y. All steps are chosen independently, so we can consider movement in the x and y directions separately. Let $\mathcal{C}$ be the event that the a_1 and a_2 collide in round $r + m$, $\mathcal{C}_x$ be the event that they have the same x position, and $\mathcal{C}_y$ be the event that they have the same y position. We have:

$$\mathbb{P}\left[\mathcal{C} | M_x = m_x, M_y = m_y\right] = \mathbb{P}\left[\mathcal{C}_x | M_x = m_x\right] \cdot \mathbb{P}\left[\mathcal{C}_y | M_y = m_y\right]. \qquad [3]$$

We first consider $\mathbb{P}\left[\mathcal{C}_x | M_x = m_x\right]$. All bounds will hold symmetrically for the y dimension. We split our analysis into two cases. Let $\mathcal{C}_x^1$ be the event that the two agents have the same x position after round $r + m$ and have identical displacements from their starting locations. Let $\mathcal{C}_x^2$ be the event that the two agents have the same x position after round $r + m$ but *do not* have identical displacements.

This requires that the agents 'wrap around' the torus, ending at the same position despite moving different amounts in the x direction. We have $\mathbb{P}[\mathcal{C}_x|M_x = m_x] = \mathbb{P}[\mathcal{C}_x^1|M_x = m_x] + \mathbb{P}[\mathcal{C}_x^2|M_x = m_x]$.

Let m_x^i be the number of steps that agent a_i takes in the x direction. We can write the displacement of agent i as $\sum_{j=1}^{m_x^i} s_j^i$ where s_j^i the direction of the agent's j^{th} step in the x direction. Each s_j^i is an independent random variable equal to 1 with probability $1/2$ and -1 with probability $1/2$. With this notation we see that $\mathcal{C}_x^1$ is the event that $\sum_{j=1}^{m_x^1} s_j^1 - \sum_{j=1}^{m_x^2} s_j^2 = 0$. Since each s_j^2 is equal to ± 1 with equal probability, s_j^2 is identically distributed to $-s_j^2$. We thus have:

$$\mathbb{P}[\mathcal{C}_x^1|M_x = m_x] = \mathbb{P}\left[\left(\sum_{j=1}^{m_x^1} s_j^1 - \sum_{j=1}^{m_x^2} s_j^2 = 0\right)|M_x = m_x\right] = \mathbb{P}\left[\left(\sum_{j=1}^{m_x^1} s_j^1 + \sum_{j=1}^{m_x^2} s_j^2 = 0\right)|M_x = m_x\right]$$
$$= \mathbb{P}\left[\left(\sum_{j=1}^{m_x} t_j = 0\right)|M_x = m_x\right].$$

where each t_j is an independent random variable equal to 1 with probability $1/2$ and -1 otherwise. The above probability is identical to the probability that a single random walk takes m_x steps and has 0 overall displacement – that is, that it takes an equal number of clockwise and counterclockwise steps. It can be computed as:

$$\mathbb{P}[\mathcal{C}_x^1|M_x = m_x] = \binom{m_x}{m_x/2}\left(\frac{1}{2}\right)^{m_x} = \frac{m_x!}{(\frac{m_x}{2}!)^2} \cdot \left(\frac{1}{2}\right)^{m_x}. \quad [4]$$

Above we assume m_x is even – otherwise $\mathcal{C}_x^1$ cannot occur. By Stirling's approximation for any $n > 0$, $n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n\left(1 + O\left(\frac{1}{n}\right)\right)$. Plugging this into 4:

$$\mathbb{P}[\mathcal{C}_x^1|M_x = m_x] = \frac{m_x!}{(\frac{m_x}{2}!)^2} \cdot \left(\frac{1}{2}\right)^{m_x} = \Theta\left(\frac{1}{\sqrt{m_x + 1}}\right).$$

We use $m_x + 1$ instead of m_x in the denominator so that the bound holds in the case when $m_x = 0$.

We next bound $\mathbb{P}\left[\mathcal{C}_x^2|M_x = m_x\right]$ – the probability that two agents have the same x position after round $r + m$ but have different total displacements. In order to have the same position by different displacements, the agent's displacements must differ by an nonzero integer multiple of $\sqrt{A}$ – the side length of the torus. We can thus write, letting $\mathbb{Z} \setminus 0$ denote the set of nonzero integers:

$$\begin{aligned}\mathbb{P}\left[\mathcal{C}_x^2|M_x = m_x\right] &= \sum_{c \in \mathbb{Z}\setminus 0} \mathbb{P}\left[\left(\sum_{j=1}^{m_x^1} s_j^1 - \sum_{j=1}^{m_x^2} s_j^2 = c\sqrt{A}\right)|M_x = m_x\right]\\ &= \sum_{c \in \mathbb{Z}\setminus 0} \mathbb{P}\left[\left(\sum_{j=1}^{m_x^1} s_j^1 + \sum_{j=1}^{m_x^2} s_j^2 = c\sqrt{A}\right)|M_x = m_x\right]\\ &= \sum_{c \in \mathbb{Z}\setminus 0} \mathbb{P}\left[\left(\sum_{j=1}^{m_x} t_j = c\sqrt{A}\right)|M_x = m_x\right]\\ &= 2\sum_{c=1}^{\left\lfloor \frac{m_x}{\sqrt{A}} \right\rfloor} \mathbb{P}\left[\left(\sum_{j=1}^{m_x} t_j = c\sqrt{A}\right)|M_x = m_x\right] \end{aligned} \quad [5]$$

where again each t_j is an independent random variable equal to 1 with probability $1/2$ and -1 otherwise. In the second step we again used that s_j^2 is identically distributed to $-s_j^2$. In the last step we used that t_j is identically distributed to $-t_j$ and so the probability of the displacement being $c\sqrt{A}$ is identical to the probability of it being $-c\sqrt{A}$. Additionally, it suffices to consider $c \le \left\lfloor \frac{m_x}{\sqrt{A}} \right\rfloor$ since otherwise the total displacement after m_x steps cannot possibly be $c\sqrt{A} > m_x$.

The above is identical to the probability that a single m_x step random walk has overall displacement $\pm c\sqrt{A}$ for some integer $c \ge 1$ (and so 'wraps around' the torus, ending at its starting location). Roughly, we will bound the probability of this event by the probability that the random walk ends at *any* other location on the torus. There are $\sqrt{A}$ such locations, so the probability is bounded by $O\left(\frac{1}{\sqrt{A}}\right)$. Formally, we have, using Eq. (5):

$$\mathbb{P}\left[\mathcal{C}_x^2|M_x = m_x\right] = 2\sum_{c=1}^{\left\lfloor \frac{m_x}{\sqrt{A}} \right\rfloor} \left(\frac{1}{2}\right)^{m_x} \cdot \binom{m_x}{\frac{m_x - c\sqrt{A}}{2}}, \quad [6]$$

If $\frac{m_x - c\sqrt{A}}{2}$ is not an integer, we use the convention that the binomial coefficient equals 0.

For $i \in [1, ..., \sqrt{A} - 1]$, let $\mathcal{D}_x^i$ be the event that a single random walk is i steps clockwise from its starting location after taking M_x

steps. We have:

$$\begin{aligned}
\mathbb{P}[\mathcal{D}_x^i | M_x = m_x] &= \left(\frac{1}{2}\right)^{m_x} \cdot \sum_{c=-\left\lfloor \frac{m_x+i}{\sqrt{A}} \right\rfloor}^{\left\lfloor \frac{m_x-i}{\sqrt{A}} \right\rfloor} \binom{m_x}{\frac{m_x+i+c\sqrt{A}}{2}} \\
&\geq \left(\frac{1}{2}\right)^{m_x} \cdot \sum_{c=-\left\lfloor \frac{m_x+i}{\sqrt{A}} \right\rfloor}^{-1} \binom{m_x}{\frac{m_x+i+c\sqrt{A}}{2}} \\
&\geq \left(\frac{1}{2}\right)^{m_x} \cdot \sum_{c=1}^{\left\lfloor \frac{m_x}{\sqrt{A}} \right\rfloor} \binom{m_x}{\frac{m_x+i-c\sqrt{A}}{2}}. \qquad [7]
\end{aligned}$$

For any $i \in [1, ..., \sqrt{A}-1]$, and any $c \geq 1$, $\frac{m_x+i-c\sqrt{A}}{2}$ is closer to $\frac{m_x}{2}$ than $\frac{m_x-c\sqrt{A}}{2}$ is, so

$$\binom{m_x}{\frac{m_x+i-c\sqrt{A}}{2}} > \binom{m_x}{\frac{m_x-c\sqrt{A}}{2}} \qquad [8]$$

as long as $\frac{m_x+i-c\sqrt{A}}{2}$ is an integer. This allows us to lower bound $\mathbb{P}[\mathcal{D}_x^i | M_x = m_x]$ using $\mathbb{P}\left[\mathcal{C}_x^2 | M_x = m_x\right]$. Let $\mathcal{E}_{i,c}$ equal 1 if $\frac{m_x+i-c\sqrt{A}}{2}$ is an integer and 0 otherwise. Since $\mathcal{C}_x^2$ and each $\mathcal{D}_x^i$ are disjoint events:

$$\mathbb{P}\left[\mathcal{C}_x^2 | M_x = m_x\right] + \sum_{i=1}^{\sqrt{A}-1} \mathbb{P}\left[\mathcal{D}_x^i | M_x = m_x\right] \leq 1$$

and so applying Eq. (7),

$$\mathbb{P}\left[\mathcal{C}_x^2 | M_x = m_x\right] + \left(\frac{1}{2}\right)^{m_x} \cdot \sum_{i=1}^{\sqrt{A}-1} \left(\sum_{c=1}^{\left\lfloor \frac{m_x}{\sqrt{A}} \right\rfloor} \binom{m_x}{\frac{m_x+i-c\sqrt{A}}{2}} \right) \leq 1.$$

Using Eq. (8) and switching summations we then have:

$$\mathbb{P}\left[\mathcal{C}_x^2 | M_x = m_x\right] + \left(\frac{1}{2}\right)^{m_x} \cdot \sum_{c=1}^{\left\lfloor \frac{m_x}{\sqrt{A}} \right\rfloor} \left(\binom{m_x}{\frac{m_x-c\sqrt{A}}{2}} \cdot \sum_{i=1}^{\sqrt{A}-1} \mathcal{E}_{i,c} \right) \leq 1$$

and so finally $\mathbb{P}\left[\mathcal{C}_x^2 | M_x = m_x\right] \cdot \Theta(\sqrt{A}) \leq 1$ by Eq. (6) with the fact that $\sum_{i=1}^{\sqrt{A}-1} \mathcal{E}_{i,c} = \Theta\left(\sqrt{A}\right)$ for all c since $\frac{m_x+i-c\sqrt{A}}{2}$ is integral for half the possible $i \in [1, ..., \sqrt{A}-1]$. Rearranging, we have $\mathbb{P}\left[\mathcal{C}_x^2 | M_x = m_x\right] = O\left(\frac{1}{\sqrt{A}}\right)$.

Combining our bounds for $\mathcal{C}_x^1$ and $\mathcal{C}_x^2$, $\mathbb{P}\left[\mathcal{C}_x | M_x = m_x\right] = \Theta\left(\frac{1}{\sqrt{m_x+1}}\right) + O\left(\frac{1}{\sqrt{A}}\right)$. Identical bounds hold for the y direction and by Eq. (3) we have:

$$\mathbb{P}\left[\mathcal{C} | M_x = m_x, M_y = m_y\right] = \Theta\left(\frac{1}{\sqrt{(m_x+1)(m_y+1)}}\right) + O\left(\frac{1}{\sqrt{A(m_x+1)}} + \frac{1}{\sqrt{A(m_y+1)}}\right) + O\left(\frac{1}{A}\right). \qquad [9]$$

Our final step is to remove the conditioning on M_x and M_y. Since direction is chosen independently and uniformly at random for each step, $\mathbb{E}\, M_x = \mathbb{E}\, M_y = m$. By a standard Chernoff bound:

$$\mathbb{P}[M_x \leq m/2] \leq 2e^{-(1/2)^2 \cdot m/2} = O\left(\frac{1}{m+1}\right).$$

(Again using $m+1$ instead of m to cover the $m = 0$ case). An identical bound holds for M_y, and so, except with probability $O\left(\frac{1}{m+1}\right)$ both are $\geq m/2$. Plugging into Eq. (9) this gives us:

$$\begin{aligned}
\mathbb{P}\left[\mathcal{C}\right] &= \Theta\left(\frac{1}{m+1}\right) + O\left(\frac{1}{\sqrt{A(m+1)}}\right) + O\left(\frac{1}{A}\right) \\
&= \Theta\left(\frac{1}{m+1}\right) + O\left(\frac{1}{A}\right).
\end{aligned}$$

□

We now give a proof of Lemma 4, which bounds the probability that two agents collide at least once.

Lemma 4 (First Collision Probability)**.** *Assuming $t \leq A$, for all $j \in [1, ..., n]$, $\mathbb{P}\left[c_j \geq 1\right] = \Theta\left(\frac{t}{A \log 2t}\right)$.*

Proof. Using the fact that c_j is identically distributed for all j,

$$\mathbb{E}\,\tilde{d} = d = \frac{1}{t} \cdot \mathbb{E} \sum_{i=1}^{n} c_i = \frac{n}{t} \cdot \mathbb{E}\, c_j = \frac{n}{t} \cdot \mathbb{P}\left[c_j \geq 1\right] \cdot \mathbb{E}[c_j | c_j \geq 1]$$

$$\frac{n}{A} = \frac{n}{t} \cdot \mathbb{P}\left[c_j \geq 1\right] \cdot \mathbb{E}[c_j | c_j \geq 1].$$

Rearranging gives:

$$\mathbb{P}\left[c_j \geq 1\right] = \frac{t}{A \cdot \mathbb{E}[c_j | c_j \geq 1]}. \qquad [10]$$

To compute $\mathbb{E}[c_j | c_j \geq 1]$, we use Lemma 3 and linearity of expectation. Since $t \leq A$, the $O\left(\frac{1}{A}\right)$ term in Lemma 3 is absorbed into the $\Theta\left(\frac{1}{m+1}\right)$. Let $r \leq t$ be the first round that the two agents collide. We have:

$$\mathbb{E}[c_j | c_j \geq 1] = \sum_{m=0}^{t-r} \Theta\left(\frac{1}{m+1}\right) = \Theta\left(\log(2(t-r+1))\right). \qquad [11]$$

After any round the agents are located at uniformly and independently chosen positions, so collide with probability exactly $1/A$. So, the probability of the *first* collision between the agents being in a given round can only decrease as the round number increases. So, at least $1/2$ of the time that $c_j \geq 1$, there is a collision in the first $t/2$ rounds (Note that we can assume $t \geq 2$ since if $t = 1$ we already have $\mathbb{E}[c_j | c_j \geq 1] = 1$. So, overall, by Eq. (11), $\mathbb{E}[c_j | c_j \geq 1] = \Theta\left(\log(2(t - t/2 + 1)\right)) = \Theta\left(\log 2t\right)$. Using Eq. (10), $\mathbb{P}\left[c_j \geq 1\right] = \Theta\left(\frac{t}{A \cdot \log 2t}\right)$, completing the proof. □

Finally, we combine the results of Lemma 3 and 4 to give our collision moment bound.

Lemma 5 (Collision Moment Bound). *For $j \in [1, ..., n]$, let $\bar{c}_j \overset{\text{def}}{=} c_j - \mathbb{E}\, c_j$ and assume $t \leq A$. There is some fixed constant w such that for any integer $k \geq 2$,*

$$\mathbb{E}\left[\bar{c}_j^k\right] \leq \frac{tw^k}{A} \cdot k! \log^{k-1}(2t).$$

Proof. We expand $\mathbb{E}[\bar{c}_j^k] = \mathbb{P}[c_j \geq 1] \cdot \mathbb{E}[\bar{c}_j^k | c_j \geq 1] + \mathbb{P}[c_j = 0] \cdot \mathbb{E}[\bar{c}_j^k | c_j = 0]$, and so by Lemma 4:

$$\mathbb{E}\left[\bar{c}_j^k\right] = O\left(\frac{t}{A \log 2t} \cdot \mathbb{E}\left[\bar{c}_j^k | c_j \geq 1\right] + \mathbb{E}\left[\bar{c}_j^k | c_j = 0\right]\right).$$

$\mathbb{E}\left[\bar{c}_j^k | c_j = 0\right] = \left(\mathbb{E}\, c_j\right)^k = (t/A)^k$. Further since $t \leq A$ by assumption, $t/A \leq 1$ and we can loosely bound $(t/A)^k \leq \frac{t}{A} k! \log^{k-1} 2t$ for all $k \geq 2$. Further, $\mathbb{E}\left[\bar{c}_j^k | c_j \geq 1\right] \leq \mathbb{E}\left[c_j^k | c_j \geq 1\right]$, since $\mathbb{E}\, c_j = \frac{t}{A} \leq 1$. So to prove the lemma, it just remains to show that $\mathbb{E}\left[c_j^k | c_j \geq 1\right] \leq k! w^k \log^k 2t$ for some w.

Conditioning on $c_j \geq 1$, we know the agents have an initial collision in some round $t' \leq t$. We split c_j over rounds as $c_j = \sum_{r=t'}^{t} c_j(r) \leq \sum_{r=t'}^{t'+t-1} c_j(r)$. To simplify notation we relabel round t' round 1 and so round $t' + t - 1$ becomes round t. After this relabeling we have $c_j(1) = 1$. This relabeling is valid since the distribution over future collisions between two agents that collide in round t' is identical, no matter the value of t'. Expanding c_j^k out fully using the summation:

$$\mathbb{E}\left[c_j^k\right] = \mathbb{E}\left[\sum_{r_1=1}^{t} \sum_{r_2=1}^{t} ... \sum_{r_k=1}^{t} c_j(r_1) c_j(r_2) ... c_j(r_k)\right]$$

$$= \sum_{r_1=1}^{t} \sum_{r_2=1}^{t} ... \sum_{r_k=1}^{t} \mathbb{E}\left[c_j(r_1) c_j(r_2) ... c_j(r_k)\right].$$

$\mathbb{E}\left[c_j(r_1) c_j(r_2) ... c_j(r_k)\right]$ is just the probability that the two agents collide in each of rounds $r_1, r_2, ..., r_k$. Assume without loss of generality that $r_1 \leq r_2 \leq ... \leq r_k$. By Lemma 3 and the fact that $c_j(1) = 1$, for some fixed w we can bound this probability $\leq \frac{w^k}{r_1(r_2 - r_1 + 1)(r_3 - r_2 + 1)...(r_k - r_{k-1} + 1)}$. Here we use the assumption that $t \leq A$ so the $O\left(\frac{1}{A}\right)$ term is absorbed into the $\Theta\left(\frac{1}{m+1}\right)$ term in Lemma 3. We then rewrite, by linearity of expectation:

$$\mathbb{E}\left[c_j^k\right] \leq k! \sum_{r_1=1}^{t} ... \sum_{r_k=r_{k-1}}^{t} \frac{w^k}{r_1(r_2 - r_1 + 1)...(r_k - r_{k-1} + 1)}.$$

The $k!$ comes from the fact that in this sum we have only ordered k-tuples and so need to multiple by $k!$ to account for the fact that the original sum is over unordered k-tuples. We can bound:

$$\sum_{r_k=r_{k-1}}^{t} \frac{1}{r_k - r_{k-1} + 1} = 1 + \frac{1}{2} + ... + \frac{1}{t} = O(\log 2t),$$

so rearranging the sum and simplifying gives:

$$\mathbb{E}\left[c_j^k\right] \le k!w^k \sum_{r_1=1}^{t} \frac{1}{r_1} \sum_{r_2=r_1}^{t} \frac{1}{r_2 - r_1 + 1} \dots \sum_{r_k=r_{k-1}}^{t} \frac{1}{r_k - r_{k-1} + 1}$$

$$\le k!w^k \sum_{r_1=1}^{t} \dots \sum_{r_{k-1}=r_{k-2}}^{t} \frac{1}{r_{k-2} - r_{k-1} + 1} \cdot O(\log 2t).$$

We repeat this argument for each level of summation replacing $\sum_{r_i=r_{i-1}}^{t} \frac{1}{r_i - r_{i-1}+1}$ with $O(\log 2t)$. Iterating through the k levels gives $\mathbb{E}\left[c_j^k\right] \le k!w^k \log^k 2t$, after w is adjusted using the constant in the $O(\log 2t)$ term, establishing the lemma. □

S3: Equalization Probability and Moment Bounds for Single Random Walks

Our analysis of the re-collision probabilities for two randomly walking agents given in Section 2 extends easily to bounds on the number of equalizations (returns to origin) of a single random walk, which may be of independent interest. We first bound the equalization probability of a walk after m steps, analogous to the two agent re-collision probability bound of Lemma 3.

Corollary 15 (Equalization Probability Bound)**.** *Consider agent a_1 randomly walking on a two-dimensional torus of dimensions $\sqrt{A} \times \sqrt{A}$. If a_1 is located at position p after round r, for any even $m \ge 0$, the probability that a_1 is again at position p after round $r+m$ is $\Theta\left(\frac{1}{m+1}\right) + O\left(\frac{1}{A}\right)$. For odd m the probability is 0.*

Proof. The analysis of Lemma 3 treats the two walks of a_1 and a_2 as a single walk with 2m total steps. An identical analysis where $2m$ is replaced by m yields the corollary. □

We next extend Lemmas 4 and 5 to a single random walk, giving

Corollary 16 (Random Walk Visits Moment Bound)**.** *Consider an agent a_1 randomly walking on a two-dimensional $\sqrt{A} \times \sqrt{A}$ torus that is initially located at a uniformly random location and takes $t \le A$ steps. Let c_j be the number of times that a_1 visits node j. There exists a fixed constant w such that for all $j \in [1, \dots A]$ and all $k \ge 2$,*

$$\mathbb{E}\left[\bar{c}_j^k\right] \le \frac{tw^k}{A} \cdot k! \log^{k-1}(2t).$$

Proof. This follows from noting that the expected number of visits to a given node is t/A and so Lemma 4 can be used in conjunction with Corollary 15 to show that $\mathbb{P}[c_j \ge 1] = \Theta\left(\frac{t}{A \log 2t}\right)$. We can then just follow the proof of Lemma 5, using Corollary 15 where needed to obtain the result. □

Corollary 17 (Equalization Moment Bound)**.** *Consider an agent a_1 randomly walking on a two-dimensional $\sqrt{A} \times \sqrt{A}$ torus. If a_1 takes $t \le A$ steps and c is the number of times it returns to its starting position (the number of equalizations), there exists a fixed constant w such that for all $k \ge 2$, $\mathbb{E}\left[\bar{c}^k\right] \le k!w^k \log^k(2t)$.*

Proof. This follows from the proof of the moment bound given in Lemma 5 for the number of collisions between two agents that are assumed to collide at least once: $\mathbb{E}[c_j^k | c_j \ge 1] \le k!w^k \log^k(2t)$. We simply replace the application of Lemma 3 with Corollary 15. □

S4: Full Proofs for Extensions to Other Topologies

We now present a number of the missing proofs for our extensions to other graph topologies in Section 3.

From Re-collision Bounds to Accurate Density Estimation. We begin with our general lemma for converting collision probability bounds to density estimation accuracy.

Lemma 8 (Re-collision Probability to Density Estimation Accuracy)**.** *Consider a regular graph with A nodes such that, if two randomly walking agents a_1 and a_2 collide in round r, for any $0 \le m \le t$, the probability that they collide again in round $r+m$ is $\Theta(\beta(m))$ for some non-increasing function $\beta(m)$. Let $B(t) \overset{\text{def}}{=} \sum_{m=0}^{t} \beta(m)$. After running for $t \le A$ steps, Algorithm 1 returns $\tilde{d}$ such that, for any $\delta > 0$, with probability $\ge 1 - \delta$, $\tilde{d} \in [(1-\epsilon)d, (1+\epsilon)d]$ for $\epsilon = O\left(\sqrt{\frac{\log(1/\delta)B(t)}{td}}\right)$.*

Proof. $\mathbb{E}\,\tilde{d} = d$ (Lemma 2) still holds as the regularity of the graph ensures that agents remain uniformly distributed on the nodes in every round (the stable distribution of any regular graph is the uniform distribution). Lemma 4 is also analogous except that Eq. (11) becomes:

$$\mathbb{E}[c_j | c_j \ge 1] = \Theta\left(\sum_{m=0}^{t-r} \beta(m)\right)$$

and using the fact that at least 1/2 the time that $c_j \ge 1$, there is a collision in the first $t/2$ rounds and that $\beta(m)$ is non-increasing, $\mathbb{E}[c_j | c_j \ge 1] = \Theta\left(\sum_{m=0}^{t/2} \beta(m)\right) = \Theta\left(B(t)\right)$. This gives:

$$\mathbb{P}[c_j \ge 1] = \Theta\left(\frac{t}{A \cdot B(t)}\right).$$

Following the moment calculations in Lemma 5, $\mathbb{E}[c_j^k | c_j \geq 1] \leq k! w^k B(t)^k$ for some constant w and hence:

$$\mathbb{E}[\bar{c}_j^k] \leq \frac{tw^k}{A} \cdot k! B(t)^{k-1}.$$

As in Corollary 6, this gives that $\sum_{j=1}^n c_j$ satisfies the Bernstein condition

$$\mathbb{E}\left[\left(\sum_{j=1}^n c_j - \mathbb{E}\left[\sum_{j=1}^n c_j\right]\right)^k\right] \leq \frac{1}{2} k! \sigma^2 b^{k-2}$$

for $b = \Theta(B(t))$ and $\sigma^2 = \Theta\left(n \cdot \frac{tB(t)}{A}\right) = \Theta\left(tdB(t)\right)$. Plugging into Lemma 7 gives $\frac{\epsilon^2 td}{B(t)} = \Theta(\log(1/\delta))$. Rearranging yields the result. □

Applying the above bound requires a constant factor approximation to the re-collision probability – the probability is $\Theta(\beta(m))$. Sometimes however, it is much easier (for example, in our proofs for k-dimensional tori, expander graphs, and hypercubes) to give just an upper bound – so the probability is $O(\beta(m))$. In this case a slightly weaker bound holds:

Lemma 18 (Re-collision Probability Upper Bound to Density Estimation Accuracy)**.** *Consider a regular graph with A nodes such that, if two randomly walking agents a_1 and a_2 collide in round r, for any $0 \leq m \leq t$, the probability that they collide again in round $r+m$ is $O(\beta(m))$ for some non-increasing function $\beta(m)$. Let $B(t) \overset{\text{def}}{=} \sum_{m=0}^t \beta(m)$. After running for $t \leq A$ steps, Algorithm 1 returns $\tilde{d}$ such that, for any $\delta > 0$, with probability $\geq 1 - \delta$ $\tilde{d} \in [(1-\epsilon)d, (1+\epsilon)d]$ for $\epsilon = O\left(\sqrt{\frac{\log(1/\delta) \cdot B(t)^2}{td}}\right)$.*

Proof. The proof is identical to that of Lemma 8 except that, we can only show $\mathbb{P}[c_j \geq 1] = O\left(\frac{t}{A}\right)$. Therefore, our moment bound becomes:

$$\mathbb{E}[\bar{c}_j^k] \leq \frac{tw^k}{A} \cdot k! B(t)^k.$$

for some constant w. This gives that $\sum_{j=1}^n c_j$ satisfies the Bernstein condition with parameters $b = \Theta(B(t))$ and $\sigma^2 = \Theta(tdB(t)^2)$. Following Lemma 8 we therefore have $\frac{\epsilon^2 td}{B(t)^2} = \Theta(\log(1/\delta))$. Rearranging yields the proof. □

Re-collision Probability Bounds for General Topologies.

Lemma 10 (Re-collision Probability Bound – High-Dimensional Torus)**.** *If two randomly walking agents a_1 and a_2 are located on a k-dimensional torus with A nodes, and collide in round r, for any constant $k \geq 3$, $m \geq 0$, the probability that a_1 and a_2 collide in round $r+m$ is $\Theta\left(\frac{1}{(m+1)^{k/2}}\right) + O\left(\frac{1}{A}\right)$.*

Proof. We closely follow the proof of Lemma 3. In total, a_1 and a_2 take $2m$ steps: M_i in each dimension for $i \in [1, ..., k]$. Let $\mathcal{C}_i$ be the event that the agents have the same position in the i^{th} dimension in round $r+m$. By the analysis of Lemma 3,

$$\mathbb{P}[\mathcal{C}_i | M_i = m_i] = \Theta\left(\frac{1}{\sqrt{m_i + 1}}\right) + O\left(\frac{1}{A^{1/k}}\right).$$

So,

$$\mathbb{P}[\mathcal{C} | M_1 = m_1, ..., M_k = m_k] = \left[\Theta\left(\frac{1}{\sqrt{m_1 + 1}}\right) + O\left(\frac{1}{A^{1/k}}\right)\right] \cdot .... \cdot \left[\Theta\left(\frac{1}{\sqrt{m_k + 1}}\right) + O\left(\frac{1}{A^{1/k}}\right)\right]. \qquad [12]$$

In expectation, $M_i = 2m/k$. So by a Chernoff bound,

$$\mathbb{P}[M_i \leq m/k] \leq 2e^{-(1/2)^2 \cdot 2m/3k} = O\left(\frac{1}{(m+1)^{k/2}}\right)$$

again assuming k is a small constant. Union bounding over all k dimensions, we have $M_i \geq m/k$ for all i except with probability $O\left(\frac{1}{(m+1)^{k/2}}\right)$ and hence by Eq. (12):

$$\mathbb{P}[\mathcal{C}] = O\left(\frac{1}{(m+1)^{k/2}}\right) + \left[\Theta\left(\frac{1}{\sqrt{m/k + 1}}\right) + O\left(\frac{1}{A^{1/k}}\right)\right]^k = \Theta\left(\frac{1}{(m+1)^{k/2}}\right) + O\left(\frac{1}{A}\right),$$

giving the lemma (again, asymptotic notation hides multiplicative factors in k since it is constant). □

Lemma 11 (Re-collision Probability Bound – Regular Expander)**.** *Let G be a k-regular expander with A nodes and adjacency matrix $\mathbf{M}$. Let $\mathbf{W} = \frac{1}{k} \cdot \mathbf{M}$ be its random walk matrix, with eigenvalues $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_A$. Let $\lambda = \max\{|\lambda_2|, |\lambda_A|\} < 1$. If two randomly walking agents a_1 and a_2 collide in round r, for any $m \geq 0$, the probability that they collide again in round $r+m$ is at most $\lambda^m + 2/A$.*

Proof. Suppose that a_1 and a_2 collide at node i in round r. The probability they re-collide at round $r+m$ is $||\mathbf{W}^m \mathbf{e}_i||_2^2$, where $\mathbf{e}_i$ is the i^{th} standard basis vector. This follows from noting that for each j, $\mathbf{W}_{i,j}^m = (\mathbf{W}^m \mathbf{e}_i)_j$ is the probability an agent is at node j after round $r+m$ given that it is at node i after round r. Since the agents move independently, both starting from node i, the probability that they both end at node j in round $r+m$ is $(\mathbf{W}^m \mathbf{e}_i)_j^2$. Summing over all possible ending positions, we thus have the re-collision probability equal to $\sum_{j=1}^A (\mathbf{W}^m \mathbf{e}_i)_j^2 = ||\mathbf{W}^m \mathbf{e}_i||_2^2$.

We bound this norm using the following lemma on how rapidly an expander random walk converges to its stable distribution:

Lemma 19 (See (6))**.** *Let G be a k-regular expander with A nodes, adjacency matrix $\mathbf{M}$, and random walk matrix $\mathbf{W} = \frac{1}{k} \cdot \mathbf{M}$. Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_A$ be the eigenvalues of $\mathbf{W}$ and $\lambda = \max\{|\lambda_2|, |\lambda_A|\} < 1$. For each $1 \leq j \leq n$,*

$$\left|(\mathbf{W}^m \cdot e_i)_j - \frac{1}{A}\right| \leq \lambda^m.$$

Now we can bound $||\mathbf{W}^m \mathbf{e}_i||_2^2$ by:

$$||\mathbf{W}^m \mathbf{e}_i||_2^2 = \sum_{j=1}^{A} (\mathbf{W}^m \mathbf{e}_i)_j^2 = \sum_{j=1}^{A} \left(\frac{1}{A} + \chi_j\right)^2,$$

where $\chi_j \overset{\text{def}}{=} (\mathbf{W}^m \cdot \mathbf{e}_i)_j - \frac{1}{A}$ so that $\chi_j \in [-1/A, \lambda^m]$ due to Lemma 19. We have $\sum_j \chi_j = \sum_j (\mathbf{W}^m \mathbf{e}_i)_j - A \cdot (1/A) = 0$. Therefore,

$$\begin{aligned} ||\mathbf{W}^m \mathbf{e}_i||_2^2 &= \sum_{j=1}^{A} \left(\frac{1}{A} + \chi_j\right)^2 \\ &= \sum_{j=1}^{A} \left(\left(\frac{1}{A}\right)^2 + \frac{2\chi_j}{A} + \chi_j^2\right) = \frac{1}{A} + \sum_{j=1}^{A} \chi_j^2. \end{aligned}$$

$\sum_j \chi_j^2$ is maximized when the number of possible j with $\chi_j = \lambda^m$ is maximized. Let $S \subset [1, A]$ be the indices j with $\chi_j = \lambda^m$. Since $\sum_j \chi_j = 0$, we have $\sum_{j \in S} \lambda^m + \sum_{j \notin S} \chi_j = 0$. Therefore, $|S| \cdot \lambda^m \leq -\sum_{j \notin S} \chi_j$. Further, since $\chi_j \in [-1/A, \lambda^m]$ we have:

$$-\sum_{j \notin S} \chi_j \leq |j \notin S| \cdot 1/A = \frac{A - |S|}{|A|} = 1 - \frac{|S|}{A}.$$

So overall, $|S| \leq \frac{1}{\lambda^m + 1/A}$. Therefore,

$$\begin{aligned} \sum_{j=1}^{A} \chi_j^2 &\leq \sum_{j \in S} \lambda^{2m} + \sum_{j \notin S} \chi_j^2 \\ &\leq \frac{\lambda^{2m}}{\lambda^m + 1/A} + \frac{A - |S|}{A^2} \leq \lambda^m + 1/A. \end{aligned}$$

Thus, $||\mathbf{W}^m \mathbf{e}_i||_2^2 \leq \lambda^m + 2/A$, giving the lemma. □

Lemma 12 (Re-collision Probability Bound – k-Dimensional Hypercube)**.** *If two randomly walking agents a_1 and a_2 are located on a k-dimensional hypercube with $A = 2^k$ vertices and collide in round r, for any $m \geq 0$, the probability that a_1 and a_2 collide in round $r + m$ is $O\left((7/10)^m + \frac{1}{\sqrt{A}}\right)$.*

Proof. A node of the hypercube can be represented as a k-bit string and each random walk step seen as choosing one of the bits uniformly at random and flipping it. If a_1 and a_2 collide, for each of the bits, the total number of times a_1 and a_2 chose that bit must be even. The total number of possible ways for re-collision to occur at round $r + m$ is exactly the number of ways $2m$ flips can be placed into k buckets, where each bucket has even number of elements. This quantity is:

$$\sum_{\substack{a_1 + \ldots + a_k = 2m \\ (a_i \mod 2) \equiv 0}} \frac{(2m)!}{a_1! \cdot \ldots \cdot a_k!}.$$

This value is equal to the coefficient of x^{2m} in the exponential generating function

$$(2m)!\left(1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \ldots\right)^k = (2m)!\left(\frac{e^x + e^{-x}}{2}\right)^k = \frac{(2m)!}{2^k} \sum_{i=0}^{k} \binom{k}{i} e^{x(2i-k)}.$$

By differentiating $2m$ times, we find that the coefficient of x^{2m} is:

$$\frac{1}{2^k} \sum_{i=0}^{k} \binom{k}{i} (2i - k)^{2m} = \sum_{i=0}^{k} \left(\binom{k}{i}/2^k\right) \cdot (2i - k)^{2m}.$$

This summation is exactly $\mathbb{E}[X^{2m}]$, where X is a sum of k i.i.d. random variables each equal to 1 with probability $1/2$ and -1 otherwise. For any $c \in (0, 1]$, we can split the expectation:

$$\begin{aligned} \mathbb{E}[X^{2m}] &= \mathbb{E}[X^{2m} || X| \geq ck] \cdot \mathbb{P}[|X| \geq ck] + \mathbb{E}[X^{2m} || X| \leq ck] \cdot \mathbb{P}[|X| \leq ck] \\ &\leq k^{2m} \cdot \mathbb{P}[|X| \geq ck] + (ck)^{2m}. \end{aligned}$$

To bound the return probability, we divide this count by the the total number of possible paths taken by a_1 and a_2 in m steps, k^{2m}, giving an upper bound of:

$$\mathbb{P}[|X| \geq ck] + c^{2m}.$$

By a Hoeffding bound, $\mathbb{P}[|X| \geq ck] \leq 2e^{-c^2 k/2}$. If we set $c = \sqrt{\ln A/k} = \sqrt{\ln 2}$ then $\mathbb{P}[|X| \geq ck] \leq 1/\sqrt{A}$. So our final probability bound is:

$$\mathbb{P}[|X| \geq ck] + c^{2m} \leq \frac{1}{\sqrt{A}} + (\sqrt{\ln 2})^{2m} < \frac{1}{\sqrt{A}} + (7/10)^m,$$

yielding the lemma. Note that, by adjusting c, it is possible to trade off the terms in the above bound, giving stronger inverse dependence on A at the expense of slower exponential decay in m. □

S5: Network Size Estimation via Random Walks

We now prove Theorem 13 which bounds the performance of Algorithm 2 for network size estimation. We then discuss how to estimate the average node degree and bound the mixing time, removing the input assumptions from Algorithm 2 and completing our analysis.

Analysis of Idealized Algorithm. We start with the analysis of Algorithm 2, which is given the average degree $\overline{\deg}$ as input and random walk starting locations distributed according to the network's stable distribution.

Throughout this section, we work directly with the weighted total collision count $C = \frac{\overline{\deg} \sum c_j}{n(n-1)t}$, showing that it is close to its expectation with high probability and hence giving the accuracy bound for $\tilde{A}$. As in the density estimation case, we start by showing that C is correct in expectation.

Lemma 20. $\mathbb{E}\, C = 1/|V|$.

Proof. Let $c_j(r)$ be the number of collisions, weighted by inverse vertex degree, walk j expects to be involved in at round r. In each round all walks are at vertex v_i with probability $p_i = \frac{\deg(v_i)}{2|E|}$, so:

$$\mathbb{E}\, c_j(r) = \sum_{i=1}^{|V|} \left[\frac{\deg(v_i)}{2|E|} \cdot \frac{(n-1)\deg(v_i)}{2|E|} \cdot \frac{1}{\deg(v_i)} \right] = \frac{n-1}{4|E|^2} \sum_{i=1}^{|V|} \deg(v_i) = \frac{n-1}{2|E|}.$$

By linearity of expectation, $\mathbb{E}\, c_j = \frac{t(n-1)}{2|E|}$, $\mathbb{E} \sum c_j = \frac{tn(n-1)}{2|E|}$ and hence, $\mathbb{E}\, C = \frac{\overline{\deg}}{2|E|} = 1/|V|$. □

We now need to show concentration of C about its expectation. Let $c_{i,j}$ be the weighted collision count between walks w_i and w_j where $i \neq j$. It is possible to follow the moment bound proof of Lemma 5 and bound all moments of $c_{i,j}$. However, we cannot claim that the different $c_{i,j}$'s are independent. Hence, we cannot prove a bound analogous to Lemma 6 and employ the concentration result of Lemma 7.

Instead, we bound just second moment (the variance) of each $c_{i,j}$ and obtain our concentration results via Chebyshev's inequality. This leads to a linear rather than logarithmic dependence on the failure probability $1/\delta$. However, we note that we can simply perform $\log(1/\delta)$ estimates each with failure probability $1/3$ and return the median, which will be correct with probability $1-\delta$.

Lemma 21 (Degree Weighted Collision Variance Bound)**.** *For all* $i, j \in [1, ..., n]$ *with* $i \neq j$*, let* $\bar{c}_{i,j} \overset{\text{def}}{=} c_{i,j} - \mathbb{E}\, c_{i,j}$*.* $\mathbb{E}\left[\bar{c}_{i,j}^2\right] = O\left(\frac{t(B(t)+|V|/|E|)}{|E|}\right)$.

Proof. We can write $\mathbb{E}\, \bar{c}_{i,j}^2 = \mathbb{E}\, c_{i,j}^2 - (\mathbb{E}\, c_{i,j})^2 \leq \mathbb{E}\, c_{i,j}^2$. We can then split $c_{i,j}$ over rounds to give:

$$\mathbb{E}\, \bar{c}_{i,j}^2 \leq \mathbb{E}\left[\left(\sum_{r=1}^{t} c_{i,j}(r) \right)^2 \right] = \sum_{r=1}^{t} \mathbb{E}\left[c_{i,j}(r)^2 \right] + 2 \sum_{r=1}^{t-1} \sum_{r'=r+1}^{t} \mathbb{E}\left[c_{i,j}(r) c_{i,j}(r') \right].$$

Since the walks are in the stable distribution, and hence located at v_i in each round with probability $\frac{\deg(v_i)}{2|E|}$, we have the weighted collision $c_{i,j}(r) = \frac{1}{\deg(v_i)}$ with probability $\frac{\deg(v_i)^2}{(2|E|)^2}$. We thus have $\mathbb{E}\left[c_{i,j}(r)^2\right] = \sum_{i=1}^{|V|} \left(\frac{\deg(v_i)^2}{(2|E|)^2} \cdot \frac{1}{\deg(v_i)^2} \right)$. $\mathbb{E}\left[c_{i,j}(r) c_{i,j}(r')\right]$ can be computed similarly by summing over all pairs of vertices $\frac{1}{\deg(v)\deg(u)}$ times the probability that the agents collide at vertex v in round r and then again at vertex u in round r'. Overall this gives:

$$\begin{aligned}
\mathbb{E}\, \bar{c}_{i,j}^2 &\leq t \sum_{i=1}^{|V|} \left(\frac{\deg(v_i)^2}{(2|E|)^2} \cdot \frac{1}{\deg(v_i)^2} \right) + 2 \sum_{r=1}^{t-1} \sum_{r'=r+1}^{t} \left(\sum_{i=1}^{|V|} \left(\frac{\deg(v_i)^2}{(2|E|)^2} \cdot \frac{1}{\deg(v_i)} \cdot \sum_{j=1}^{|V|} \frac{p(v_i, v_j, r-r')^2}{\deg(v_j)} \right) \right) \\
&\leq \frac{t|V|}{4|E|^2} + 2t \sum_{m=1}^{t-1} \left(\sum_{i=1}^{|V|} \left(\frac{\deg(v_i)}{(2|E|)^2} \cdot \beta(m) \sum_{j=1}^{|V|} p(v_i, v_j, m) \right) \right)
\end{aligned}$$

where in the last step we write $r - r' = m$ and use the fact that $\beta(m) \overset{\text{def}}{=} \frac{\max_{i,j} p(v_i, v_j, m)}{\deg(v_j)}$. We have $\sum_{j=1}^{|V|} p(v_i, v_j, m) = 1$ and so can simplify the above as:

$$\begin{aligned}
\mathbb{E}\, \bar{c}_{i,j}^2 &\leq \frac{t|V|}{4|E|^2} + 2t \sum_{m=1}^{t-1} \frac{\sum_{i=1}^{|V|} \deg(v_i)}{(2|E|)^2} \cdot \beta(m) \\
&= \frac{t|V|}{4|E|^2} + 2t \sum_{m=1}^{t-1} \frac{\beta(m)}{2|E|} = O\left(\frac{t(B(t) + |V|/|E|)}{|E|} \right).
\end{aligned}$$

□

Lemma 22 (Total Collision Variance Bound). *Let* $\overline{C} = \frac{\overline{\deg}\sum_j \bar{c}_j}{n(n-1)t}$. $\mathbb{E}\left[\overline{C}^2\right] = O\left(\frac{1}{n^2 t}\cdot\frac{B(t)|E|+|V|}{|V|^2}\right)$.

Proof. $\sum_{j=1}^n \bar{c}_j = \sum_{i,j\in[1,\dots,n],i\neq j}\bar{c}_{i,j}$. We closely follow the variance calculation in (11):

$$\mathbb{E}\left[\left(\sum_{i,j\in[1,\dots,n],i\neq j}\bar{c}_{i,j}\right)^2\right] = \sum_{i,j\in[1,\dots,n],i\neq j}\left[\sum_{i',j'\in[1,\dots,n],i\neq j}\bar{c}_{i,j}\cdot\bar{c}_{i',j'}\right]$$

$$= 2\binom{n}{2}\mathbb{E}\left[\bar{c}_{i,j}^2\right] + 4!\binom{n}{4}\left(\mathbb{E}\,\bar{c}_{i,j}\right)^2 + 2\cdot 3!\binom{n}{3}\mathbb{E}\,\bar{c}_{i,j}\bar{c}_{i,k}.$$

The first term corresponds to the cases when $i = i'$ and $j = j'$. The second corresponds to $i\neq i'$ and $j\neq j'$, in which case $\bar{c}_{i,j}$ and $\bar{c}_{i',j'}$ are independent and identically distributed. The $4!\binom{n}{4}$ multiplier is the number of ways to choose an ordered set of four distinct indices. The last term corresponds to all cases when either $i = i'$ or $j = j'$. There are $3!\binom{n}{3}$ ways to choose an ordered set of three distinct indices, multiplied by two to account for the repeated index being in either the first or second position. Using $\mathbb{E}\,\bar{c}_{i,j} = 0$ and the bound on $\mathbb{E}[\bar{c}_{i,j}^2]$ from Lemma 21:

$$\mathbb{E}\left[\left(\sum_{i,j\in[1,\dots,n],i\neq j}\bar{c}_{i,j}\right)^2\right] = O\left(\frac{n^2 t(B(t)+|V|/|E|)}{|E|}\right) + 0 + 2\cdot 3!\binom{n}{3}\mathbb{E}\,\bar{c}_{i,j}\bar{c}_{i,k}. \qquad [13]$$

When $j\neq k$, $\bar{c}_{i,j}$ and $\bar{c}_{i,k}$ are independent and identically distributed conditioned on the path that walk w_i traverses. Let Ψ_i be the t step path chosen by w_i.

$$\begin{aligned}\mathbb{E}\left[\bar{c}_{i,j}\bar{c}_{i,k}\right] &= \sum_{\psi_i}\mathbb{P}\left[\Psi_i=\psi_i\right]\cdot\mathbb{E}\left[\bar{c}_{i,j}|\Psi_i=\psi_i\right]\cdot\mathbb{E}\left[\bar{c}_{i,k}\middle|\Psi_i=\psi_i\right]\\ &= \sum_{\psi_i}\mathbb{P}\left[\Psi_i=\psi_i\right]\cdot\mathbb{E}\left[\bar{c}_{i,j}|\Psi_i=\psi_i\right]^2\\ &= \sum_{\psi_i}\mathbb{P}\left[\Psi_i=\psi_i\right]\cdot\left(\mathbb{E}\left[c_{i,j}|\Psi_i=\psi_i\right]-\mathbb{E}\left[c_{i,j}\right]\right)^2. \qquad [14]\end{aligned}$$

$\mathbb{E}\left[c_{i,j}|\Psi_i=\psi_i\right] = \sum_{r=1}^t\frac{\deg(\psi_i(r))}{2|E|}\cdot\frac{1}{\deg(\psi_i(r))} = \frac{t}{2|E|} = \mathbb{E}\left[c_{i,j}\right]$. That is, the expected number of collisions is identical for every path of w_i. Plugging into Eq. (14), $\mathbb{E}\left[\bar{c}_{i,j}\bar{c}_{i,k}\right] = 0$.

So finally, plugging back into equation Eq. (13), $\mathbb{E}\left[\left(\sum_{i,j\in[1,\dots,n],i\neq j}\bar{c}_{i,j}\right)^2\right] = O\left(\frac{n^2 t(B(t)+|V|/|E|)}{|E|}\right)$ and thus:

$$\mathbb{E}\left[\overline{C}^2\right] = O\left(\frac{n^2 t(B(t)+|V|/|E|)}{|E|}\cdot\left(\frac{\overline{\deg}}{n(n-1)t}\right)^2\right) = O\left(\frac{1}{n^2 t}\cdot\frac{(B(t)+|V|/|E|)\cdot|E|}{|V|^2}\right) = O\left(\frac{1}{n^2 t}\cdot\frac{B(t)|E|+|V|}{|V|^2}\right).$$

□

With this variance bound in place, we can finally prove Theorem 13.

Proof of Theorem 13. Note that $\bar{C} = C - \mathbb{E}\,C$ and by Lemma 20, $\mathbb{E}\,C = 1/|V|$. By Chebyshev's inequality Lemma 22 gives:

$$\mathbb{P}\left[|C-\mathbb{E}\,C|\geq\epsilon\,\mathbb{E}\,C\right]\leq\frac{1}{\epsilon^2 n^2 t}\cdot(B(t)|E|+|V|).$$

Rearranging gives us that, in order to have $C\in\left[\frac{1-\epsilon}{|V|},\frac{1+\epsilon}{|V|}\right]$ with probability δ, we must have:

$$n^2 t = \Theta\left(\frac{B(t)|E|+|V|}{\epsilon^2\delta}\right).$$

Since $\tilde{A} = 1/C$, if $C\in\left[\frac{1-\epsilon}{|V|},\frac{1+\epsilon}{|V|}\right]$ then $\tilde{A}\in\left[\frac{|V|}{1+\epsilon},\frac{|V|}{1-\epsilon}\right]\subseteq[(1-2\epsilon)|V|,(1+2\epsilon)|V|]$ as long as $\epsilon < 1/2$. This gives the theorem after adjusting constants on ϵ and recalling that $\overline{\deg} = |E|/|V|$. □

Estimating The Average Degree. We now show how to estimate the value of $\overline{\deg}$ used in Algorithm 2. Specifically, we need a $(1\pm\epsilon)$ approximation to $\frac{1}{\overline{\deg}}$. If we then substitute this into the formula $\tilde{A} = \frac{\sum_j c_j}{\overline{\deg}\cdot n(n-1)t}$, we still have a $(1\pm O(\epsilon))$ approximation to the true network size. We use the algorithm and analysis of (11), which gives a simple approximation via inverse degree sampling.

Algorithm 4 Average Degree Estimation

```
input: $n$ random starting locations $[w_1,\dots,w_n]$ distributed independently according to the network's stable distribution.
  $\forall j$, set $d_j := \frac{1}{\deg(w_j)}$                         ▷ Sampling
  return $D := \frac{\sum d_j}{n}$
```

Theorem 23 (Average Degree Estimation)**.** *If* $n = \Theta\left(\frac{1}{\epsilon^2 \delta} \cdot \frac{\overline{\deg}}{\deg_{\min}}\right)$*, Algorithm 4 returns D such that, with probability at least* $1 - \delta$*,* $D \in \left[\frac{1-\epsilon}{\overline{\deg}}, \frac{1+\epsilon}{\overline{\deg}}\right]$.

Proof. Using that in the stable distribution a walk is at vertex v_i with probability $\frac{\deg(v_i)}{2|E|}$ we have:

$$\mathbb{E}\, D = \frac{1}{n}\sum_{j=1}^{n} \mathbb{E}\, d_j = \frac{1}{n} \cdot n \cdot \sum_{i=1}^{|V|}\left(\frac{\deg(v_i)}{2|E|} \cdot \frac{1}{\deg(v_i)}\right) = \frac{|V|}{2|E|} = \frac{1}{\overline{\deg}}.$$

For each d_j let $\bar{d}_j = d_j \,\mathbb{E}\, d_j$. We have $\mathbb{E}[\bar{d}_j^2] = \mathbb{E}[d_j^2] - \mathbb{E}[d_j]^2 \leq \mathbb{E}[d_j^2]$. We can explicitly compute this expectation as:

$$\mathbb{E}[d_j^2] = \sum_{i=1}^{|V|} \frac{\deg(v_i)}{2|E|} \frac{1}{\deg(v_i)^2} \leq \frac{|V|}{2|E| \deg_{\min}} = \frac{1}{\deg_{\min}} \cdot \frac{1}{\overline{\deg}}.$$

Additionally, since each d_j is independent and identically distributed, and since $\bar{D} = \frac{1}{n}\sum d_j$, letting $\bar{D} = D - \mathbb{E}\, D$,

$$\mathbb{E}\left[\bar{D}^2\right] = \frac{1}{n}\,\mathbb{E}[\bar{d}_j^2] \leq \frac{1}{n}\,\mathbb{E}[d_j^2] = \frac{1}{n \deg_{\min}} \cdot \frac{1}{\overline{\deg}}$$

Applying Chebyshev's inequality and the fact that $\mathbb{E}\, D = \frac{1}{\overline{\deg}}$: $\mathbb{P}\left[|D - \mathbb{E}\, D| \leq \frac{\epsilon}{\overline{\deg}}\right] \leq \frac{\overline{\deg}}{\epsilon^2 n \deg_{min}}$. Rearranging, to succeed with probability at least $1 - \delta$ it suffices to set $n = \Theta\left(\frac{1}{\epsilon^2 \delta} \cdot \frac{\overline{\deg}}{\deg_{\min}}\right)$. □

Handling Burn-In Error. Finally, we remove our assumption that walks start distributed exactly according to the network's stable distribution, rigorously bounding the length of burn-in required before running Algorithm 2.

Let $\mathcal{D}^* \in \mathbb{R}^{|V|^n}$ be a vector representing the true stable distribution of n random walks on G and $\mathcal{D}_t \in \mathbb{R}^{|V|^n}$ be a vector representing the distribution of the walks after running for t burn-in steps. Specifically, each walk $w_1, ..., w_n$ is initialized at a single seed vertex v. For t rounds we then update the location of each walk independently by moving to a randomly chosen neighbor. Both vectors are probability distributions: they have all entries in $[0, 1]$ and $\|\mathcal{D}^*\|_1 = \|\mathcal{D}\|_1 = 1$.

Let $\Delta = \mathcal{D}^* - \mathcal{D}_t$ and assume that $\|\Delta\|_1 \leq \delta$. We can consider two equivalent algorithms: draw an initial set of locations $W = w_1, ..., w_n$ from $\mathcal{D}^*$, run Algorithm 2, and then artificially fail with probability $\max\{0, \Delta(W)\}$. Alternatively, draw $W = w_1, ..., w_n$ from $\mathcal{D}_t$, run Algorithm 2, and then artificially fail with probability $\max\{0, -\Delta(W)\}$. These algorithms are clearly equivalent. The first obtains a good estimator with probability $1 - 2\delta$ - probability δ that Algorithm 2 fails when initialized via the stable distribution $\mathcal{D}^*$ by Theorem 13 plus an artificial failure probability of $\leq \|\Delta\|_1 \leq \delta$. The second then clearly also fails with probability 2δ. This can only be higher than if we did not perform the artificial failure after running Algorithm 2. Therefore, running Algorithm 2 with a set of random walks initially distributed according to $\mathcal{D}_t$ yields success probability $\geq 1 - 2\delta$.

How long must the burn-in period be to ensure $\|\mathcal{D}^* - \mathcal{D}_t\|_1 \leq \delta$? Let $\mathbf{W}$ be the random walk matrix of G. Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_A$ be the eigenvalues of $\mathbf{W}$ and $\lambda = \max\{|\lambda_2|, |\lambda_{|V|}|\}$. Let $\mathcal{C}_t \in \mathbb{R}^{|V|}$ denote the location distribution for a single random walk after burn-in and $\mathcal{C}^* \in \mathbb{R}^{|V|}$ denote the stable distribution of a single random walk. If we have, for all i, $|\mathcal{C}_t(v_i) - \mathcal{C}^*(v_i)| \leq \delta/n \cdot \mathcal{C}^*(v_i)$ then for any W:

$$\begin{aligned}
|\mathcal{D}_t(W) - \mathcal{D}^*(W)| &= \left|\prod_{i=1}^{n} \mathcal{C}_t(w_i) - \prod_{i=1}^{n} \mathcal{C}^*(w_i)\right| \\
&\leq \prod_{i=1}^{n} (\mathcal{C}^*(w_i) + \delta/n \cdot \mathcal{C}^*(w_i)) - \prod_{i=1}^{n} \mathcal{C}^*(w_i) \\
&< \mathcal{D}^*(W) \sum_{i=1}^{n} \binom{n}{i} (\delta/n)^i \leq \mathcal{D}^*(W) \sum_{i=1}^{n} \delta^i \leq 2\delta \cdot \mathcal{D}^*(W),
\end{aligned}$$

as long as $\delta < 1/2$. This multiplicative bound gives $\|\mathcal{D}^* - \mathcal{D}_t\|_1 \leq 2\delta$. By standard mixing time bounds ((6), Theorem 5.1), $|\mathcal{C}_t(v_i) - \mathcal{C}^*(v_i)| \leq \frac{\delta}{n|E|} \cdot \mathcal{C}^*(v_i)$ for all i after $M = O\left(\frac{\log(n|E|/\delta)}{1-\lambda}\right) = O\left(\frac{\log(|E|/\delta)}{1-\lambda}\right)$ burn-in steps (since $n < |E|$ or else we could have scanned the full graph.)