The Impact of Mobility on Gossip Algorithms

Anand Dilip Sarwate, *Member, IEEE*, and Alexandros G. Dimakis, *Member, IEEE*

***Abstract*—The influence of node mobility on the convergence time of averaging gossip algorithms in networks is studied. It is shown that a small number of fully mobile nodes can yield a significant decrease in convergence time. A method is developed for deriving lower bounds on the convergence time by merging nodes according to their mobility pattern. This method is used to show that if the agents have 1-D mobility in the same direction, the convergence time is improved by at most a constant. Upper bounds on the convergence time are obtained using techniques from the theory of Markov chains and show that simple models of mobility can dramatically accelerate gossip as long as the mobility paths overlap significantly. Simulations verify that different mobility patterns can have significantly different effects on the convergence of distributed algorithms.**

***Index Terms*—Consensus protocols, distributed algorithms, distributed averaging, distributed processing, gossip protocols, Markov chains, mobility, peer-to-peer networks, wireless sensor networks.**

I. Introduction

GOSSIP algorithms are distributed message passing schemes that are used to disseminate and process information in networks [1]. Average consensus [2]–[4] and averaging gossip algorithms [5], [6] form an important special case of schemes that can compute linear functions of the data in a robust and distributed way. Such schemes have found numerous uses for distributed estimation, localization, and optimization [7]–[9] and also for compressive sensing of sensor measurements and field estimation [11]. In this paper, we study gossip algorithms that compute linear functions and will not discuss related problems such as information dissemination (see, e.g., [12] and [13] and references therein).

Gossip algorithms are a natural fit for wireless ad hoc and sensor network applications because of their distributed and robust nature. Recently, the broadcast nature of wireless communication has been exploited to improve convergence [10], [14], [15]. Another key feature of some wireless networks is node *mobility*; to the best of our knowledge, the impact of mobility on gossip algorithms has not been significantly investigated. In this paper, we attempt to analyze how mobility can (or cannot) help the convergence of gossip algorithms. For fixed nodes in a random geometric graph (RGG) or grid (both popular model topologies for large wireless ad hoc and sensor networks), standard gossip is extremely wasteful in terms of communication requirements; even optimized standard gossip algorithms on a grid converge very slowly, requiring $\Theta(n^2 \log \epsilon^{-1})$ messages [6], [16] to compute the average within accuracy ϵ. Observe that this is of the same order as requiring every node to flood its estimate to all other nodes. The obvious solution of averaging numbers on a spanning tree and flooding back the average to all the nodes requires only $\Theta(n)$ messages. Clearly, constructing and maintaining a spanning tree in dynamic and ad hoc networks introduces significant overhead and complexity, but a quadratic number of messages is a high price to pay for fault tolerance. In this context, what kind of mobility patterns are beneficial and how many mobile agents are needed to boost the convergence speed? Our results suggest that certain kinds of mobility can, in some cases, significantly accelerate convergence. This study is a first step to understand how mobility can impact the convergence of iterative message-passing schemes, at least for the special case of pairwise averaging where the convergence behavior is better understood.

A. Main Results

Our first result is that if m nodes have full mobility and the others are fixed in a grid, the convergence time drops to $\Theta(n^2/m \log \epsilon^{-1})$. Therefore, even a vanishingly small fraction of mobile nodes can change the order of messages required for convergence. In particular, if any constant fraction of nodes have full mobility, the convergence time drops to $\Theta(n \log \epsilon^{-1})$, the same order as a fully connected graph.

Our second result is that some mobility patterns might not be beneficial. We show that even if all the nodes of the network have 1-D mobility in the same direction (e.g., horizontal), this yields no benefit in the convergence time, up to constants. Intuitively, this is because the information must still diffuse across the other direction (e.g., vertical). Finally, we show that 1-D mobility with a randomly selected direction is as good as full mobility.

In order to obtain these results, we develop a novel method for deriving *lower bounds* on the convergence time of gossip algorithms with mobile nodes by merging nodes with similar mobility regions. This method is based on a characterization of the convergence time of Markov chains in terms of a functional called the Dirichlet form [17]. Our upper bounds are derived using the so-called Poincaré inequality [18] and the related canonical path method [19]; a version of this technique

Manuscript received December 29, 2009; revised June 27, 2011; accepted September 16, 2011. Date of current version February 29, 2012. The work of A. D. Sarwate was supported by the California Institute for Telecommunications and Information Technology (CALIT2) at University of California, San Diego. The work of A. G. Dimakis was supported by the National Science Foundation Career Award CCF 1055099. The material in this paper was presented in part at INFOCOM 2009 and CAMSAP 2009.

A. D. Sarwate was with the Information Theory and Applications Center, University of California, San Diego, La Jolla, CA 92093-0447 USA. He is now with the Toyota Technological Institute at Chicago, Chicago, IL 60637 USA (e-mail: asarwate@ttic.edu).

A. G. Dimakis is with the Department of Electrical Engineering Systems Division, University of Southern California, Los Angeles, CA 90089-2560 USA (e-mail: dimakis@usc.edu).

Communicated by M. Motani, Associate Editor for Communication Networks.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2011.2177753

has also been previously used to study gossip algorithms [20]. Our techniques are fairly general; while we illustrate applications to grid networks and RGGs, the methods can be applied to other topologies.

II. Network Model and Preliminaries

A. Time Model

We use the asynchronous time model [6], [21], which is well matched to the distributed nature of wireless networks. In particular, we assume that each sensor has an independent clock whose "ticks" are distributed as a rate λ Poisson process. Our analysis is based on measuring time in terms of the number of ticks of an equivalent single virtual global clock ticking according to a rate $n\lambda$ Poisson process. An exact analysis of the time model can be found in [6]. We will refer to the time between two consecutive clock ticks as one time slot. This modeling assumption results in a discrete-time system in which one sensor is selected uniformly in each time slot.

Throughout this paper we will be analyzing the number of required messages without worrying about delay. We can, therefore, adjust the length of the time slots relative to the communication time so that only one packet exists in the network at each time slot with high probability. Note that this assumption is made only for analytical convenience; in a practical implementation, several packets might coexist in the network, but the associated issues are beyond the scope of this paper.

B. Network and Mobility Model

Suppose we have a collection of n agents $\mathcal{A}$. At the first time slot, each agent i starts at some initial location with a scalar $x_i(0)$. We will denote the vector of their initial values by $\mathbf{x}(0)$. The objective of our algorithm is for every agent to estimate the average

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i(0).$$

In order to accomplish this goal, the agents pass messages between each other to communicate their estimates. We assume that this communication always succeeds. We also assume that the messages are real numbers; the effects of message quantization in gossip and consensus algorithms is an active area of research [22]–[29].

The n agents can move in an area $\mathcal{G}$. For example, we may take $\mathcal{G}$ to be a graph with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$. Agents at locations v and v' can communicate if either $v = v'$ or $(v, v') \in \mathcal{E}$. Another example is taking $\mathcal{G}$ to be the unit square and allowing agents at v and v' to communicate if the distance $d(v, v')$ is less than some radius $r(n)$. For each location l in $\mathcal{G}$, there is a set of locations $\mathcal{N}(l) \subseteq \mathcal{G}$ such that an agent at l can communicate with agents in $\mathcal{N}(l)$. If $l' \in \mathcal{N}(l)$ then $l \in \mathcal{N}(l')$.

In this paper, we will use two networks to illustrate our results. However, the methods we describe can be used for more general networks with bidirectional communication.

1) Our first example is the $\sqrt{n} \times \sqrt{n}$ discrete lattice on the torus. The set of locations $\mathcal{V}$ is $\{0, 2, \ldots, \sqrt{n} - 1\}^2$ and there are edges between (i, j) and (i', j') if $i' = (i \pm 1) \bmod \sqrt{n}$ and $j' = (j \pm 1) \bmod \sqrt{n}$. There are n agents, one for each location in $\mathcal{V}$, and at time 0 they each occupy distinct locations in $\mathcal{V}$. For a location (i, j), we call the i the row coordinate and j the column coordinate.
2) The second example is the RGG model on the unit torus. The unit torus is formed from the unit square by "glueing" opposite edges together. The agent locations are in $[0, 1]^2$ and the initial positions of the agents are chosen uniformly in $[0, 1]^2$. Agents can communicate with each other if the distance between them on the torus is less than $r(n) = \sqrt{5c\frac{\log n}{n}}$, where $c \geq 10$ ensures some useful regularity properties [20] discussed subsequently. Again, for an agent at (i, j), we call the i the row coordinate and j the column coordinate.

Under agent-based mobility, at each time step, agent i moves to a new location in $\mathcal{G}$ chosen according to a fixed probability distribution μ_i. Therefore, the sequence of agent locations $l_i(1), l_2(t), \ldots, l_i(t)$ is independent and identically distributed (i.i.d.) according to the distribution μ_i. We call the collection of distributions $\{\mu_i : i \in \mathcal{A}\}$ an agent-based mobility pattern. Our theoretical results in this paper are for agent-based mobility. In particular, we study a few simple examples of mobility.

1) A simple example of agent-based mobility is full uniform mobility. In this model, μ_i is the uniform distribution on $\mathcal{G}$ for each $i \in \mathcal{A}$. This corresponds to the case where each agent is equiprobably at any location in the $\mathcal{G}$ at time t. This is similar to the model proposed by Grossglauser and Tse [30]. We will also consider a static network with m fully mobile agents added to the network.
2) In the *horizontal mobility* model, each agent selects a new horizontal location uniformly at each time. For the torus, it selects a new column coordinate uniformly from $\{1, 2, \ldots, \sqrt{n}\}$. For the RGG, it selects a new horizontal coordinate uniformly from $[0, 1]$.
3) In the *bidirectional* model, each agent selects equiprobably whether it will move horizontally or vertically for all time. At each time step, the horizontal agents select a new horizontal coordinate uniformly, and the vertical agents select a new vertical coordinate uniformly.
4) In a *local* model for the torus, an agent that starts initially at location (i, j) chooses a new location uniformly in the square of size $(2m + 1)^2$ centered at (i, j). That is, the horizontal coordinate is uniformly distributed in $\{i - m, \ldots, i + m\} \bmod \sqrt{n}$ and the vertical coordinate is chosen uniformly in $\{j - m, \ldots, j + m\} \bmod \sqrt{n}$. Once the new coordinates are chosen, an agent can communicate with other agents in the same or adjacent locations in the $\sqrt{n} \times \sqrt{n}$ torus.

The key assumption in all our mobility models is that in each gossip time slot, the positions of the mobile agents are selected *independently* from some distribution supported on a subregion of the space, similarly to Grossglauser and Tse [30]. Popular mobility models like the random walk model [31], [32], random waypoint model [33], and random direction model [34] have time dependences. If, however, the duration of one gossip time slot is comparable or larger than the mixing time of the mobility model, the positions of the agents will be approximately independent (see also [35]). If delay is not an issue, we can always set the duration of the gossip time slot to have that property, and in simulations we show that if we do not allow the mobility model to mix, mobility is not as helpful. We believe that our analytic results could be used to bound these more realistic mobility models, but we leave this for future work.

III. Algorithm and Main Results

A. Algorithm

The gossip algorithm that we will consider is a simple extension of the standard nearest-neighbor gossip of Boyd *et al.* [6] that includes the mobility model in a natural way. At each time step, the agents move independently to new locations. One agent is selected at random, chooses one of its neighbors according to the graph $\mathcal{G}$, and performs a pairwise average with that neighbor. More precisely, at each time $t = 1, 2, \ldots$ the following events occur.

1) Each agent $i \in \mathcal{A}$ chooses a new location $l_i(t)$ according to the mobility distribution μ_i.
2) An agent i is selected at random and selects a neighbor j uniformly from the set $\mathcal{N}(l_i(t))$. For example, if $\mathcal{G}$ is a graph, then

$$\mathcal{N}(l_i(t)) = \{k \in \mathcal{V} : (l_i(t), l_k(t)) \in \mathcal{E}.\}$$

3) The agents i and j exchange values and update their estimates

$$x_k(t) = \begin{cases} \frac{1}{2}(x_i(t-1) + x_j(t-1)) & k = i, j \\ x_k(t-1) & k \neq i, j. \end{cases}$$

Since the algorithm is randomized, we are interested in providing probabilistic bounds on its running time. Given $\epsilon > 0$, the ϵ-averaging time [6] is the earliest time at which the vector $\mathbf{x}(t)$ is ϵ close to the normalized true average with probability greater than $1 - \epsilon$

$$T_{\text{ave}}(n, \epsilon) = \sup_{x(0)} \inf_{t=0,1,2\ldots} \left\{ \mathbb{P}\left(\frac{\left\| \mathbf{x}(t) - \bar{x}\vec{1} \right\|}{\|\mathbf{x}(0)\|} \geq \epsilon \right) \leq \epsilon \right\} \quad (1)$$

where $\|\cdot\|$ denotes the Euclidean norm. Note that this is essentially measuring a rate of convergence in probability. The analysis of Denantes *et al.* [36] shows that bounds on the spectral gap yield an asymptotic deterministic rate of vanishing error. Our bounds can be used to bound both the rate of convergence in probability and to show that the averaging error decays exponentially asymptotically almost surely.

B. Main Results

Our main results characterize the benefit (or lack thereof) of mobility in speeding up the convergence of gossip algorithms. For the network on the grid or torus with no mobility, the averaging time is $T_{\text{ave}}^{(\text{torus,none})}(n, \epsilon) = \Theta(n^2 \log \epsilon^{-1})$. For the network on the RGG with the connectivity radius chosen as described previously, the averaging time is $T_{\text{ave}}^{(\text{RGG,none})}(n, \epsilon) = \Theta(\frac{n^2}{\log n} \log \epsilon^{-1})$.

1) For horizontal mobility on the RGG and the torus, the averaging time improves by at best a constant factor over the case where the agents are not mobile at all

$$T_{\text{ave}}^{(\text{torus,horiz})}(n, \epsilon) = \Omega(n^2 \log \epsilon^{-1})$$
$$T_{\text{ave}}^{(\text{RGG,horiz})}(n, \epsilon) = \Omega\left(\frac{n^2 \log \epsilon^{-1}}{\log n}\right).$$

2) For bidirectional mobility where each agent initially selects whether to move vertically or horizontally, the convergence time is within a constant factor of full mobility

$$T_{\text{ave}}^{(\text{torus,bi})}(n, \epsilon) = O\left(n \log \epsilon^{-1}\right)$$
$$T_{\text{ave}}^{(\text{RGG,bi})}(n, \epsilon) = O\left(n \log \epsilon^{-1}\right).$$

3) For n nonmobile agents on a $\sqrt{n} \times \sqrt{n}$ torus with $m < n$ agents having full mobility, the convergence time is

$$T_{\text{ave}}^{(\text{torus plus m,2D})}(n, \epsilon) = \Theta\left(\frac{n^2}{m} \log \epsilon^{-1}\right).$$

4) For the local mobility model with each agent moving in a square of size $(2m+1)^2$

$$T_{\text{ave}}^{(\text{torus,local})}(n, \epsilon) = O\left(\frac{n^2 \log m}{m^2} \log \epsilon^{-1}\right).$$

IV. Upper and Lower Bounds on Convergence Time

A. Convergence Analysis

At each step of the algorithm, the agents update their estimates of the average $\bar{x}$. Let $\mathbf{x}(t)$ denote the average estimates at time t. For agents i and j define the matrix $W^{(i,j)}$

$$W^{(i,j)} = I - \frac{1}{2}(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^T$$

where $\mathbf{e}_i$ is the vector with 1 in the ith coordinate and 0's elsewhere. If the pair (i, j) average at time t, then the new vector of averages is given by

$$\mathbf{x}(t) = W^{(i,j)} \mathbf{x}(t-1).$$

The randomness in the mobility and in the agent selection induces a probability distribution on the matrices $\{W^{(i,j)} : i, j \in \mathcal{A}\}$. Since the mobility and selection are i.i.d. across time, we can write the update as

$$\mathbf{x}(t) = \left(\prod_{s=1}^{t} W(s) \right) \mathbf{x}(0)$$

where $\{W(s)\}$ are i.i.d. random matrices. Denote the expected value of this random matrix by $\bar{W} = \mathbb{E}[W(s)]$. It is not hard to see that $\bar{W}$ is a (symmetric) stochastic matrix and, therefore, corresponds to a Markov chain. Let P_{ij} be the probability that agent i is selected in step 2 of the algorithm and it selects agent j in its neighbor set. Then, it is clear that $\mathbb{P}(W(s) = W^{(i,j)}) = P_{ij} + P_{ji}$, and that

$$\bar{W}_{ij} = \frac{1}{2}(P_{ij} + P_{ji}). \quad (2)$$

The pioneering work of Boyd *et al.* [6] showed that the convergence time of a randomized gossip algorithm is dictated by the mixing time of the Markov chain associated with $\bar{W}$. Mathematically, our problem is how to analyze the mixing time of the new graph induced by the new feature (in this case mobility) and, then, compare it to the old graph without mobility. For a Markov chain $\mathcal{M}$ with transition matrix $\bar{W}$, the convergence rate to the stationary distribution is given by $\lambda_2(\bar{W})$, the second largest eigenvalue of $\bar{W}$. Note that the largest eigenvalue $\lambda_1(\bar{W})$ is 1. Define the relaxation time T_{relax} to be the reciprocal of the spectral gap

$$T_{\text{relax}}(\bar{W}) = \frac{1}{1-\lambda_2(\bar{W})}.$$

The following theorem is implicit in [6] (see also [1]).

Theorem 1 (Convergence With T_{relax} [1], [6]): If $W = (W_{ij})$ is symmetric and n is sufficiently large, then $T_{\text{ave}}(n,\epsilon)$ is bounded by

$$T_{\text{ave}}(n,\epsilon) = \Theta\left(T_{\text{relax}}(\bar{W})\log\epsilon^{-1}\right).$$

B. Lower Bounds

In this section, we provide a general method for constructing lower bounds on the convergence time for pairwise gossip algorithms under agent-based mobility. The main intuition is to partition the set of vertices in the graph and merge all agents whose mobility is supported in the same element of the partition. This induces a transformation on the Markov chain associated with the gossip algorithm. By using an extremal characterization of the relaxation time for Markov chains, we can lower bound the relaxation time $T_{\text{relax}}(\bar{W})$ in the original gossip algorithm by that for the induced Markov chain. The only remaining issue is to choose a partition that yields a tight lower bound, which must be done by inspection. We can use this technique to show that horizontal mobility cannot improve the convergence of gossip for the torus or the RGG.

Theorem 2: Let $\{\mathcal{U}_r\}$ be any partition of the set of locations $\mathcal{G}$, and let $\hat{W}$ be the transition matrix of the chain induced by merging all agents whose mobility is restricted to a single set in the partition. Then

$$T_{\text{ave}}(n,\epsilon) = \Omega(T_{\text{relax}}(\hat{W})\log\epsilon^{-1}).$$

Proof: We begin with the set $\mathcal{G}$ on which the agents in $\mathcal{A}$ can move. Let $\{\mathcal{U}_r : r = 1,2,\ldots,M\}$ be a partition of $\mathcal{G}$. Given an agent-based mobility pattern $\{\mu_i\}$, let

$$\mathcal{C}_r = \{v \in \mathcal{A} : \mu_v(\mathcal{U}_r) = 1\}$$

be the set of agents whose mobility is restricted to $\mathcal{U}_r$. We can create a map F on the state set $\mathcal{A}$ of the Markov chain corresponding to the gossip algorithm

$$F(a) = \begin{cases} r & \text{if } a \in \mathcal{C}_r \\ a & \text{otherwise.} \end{cases}$$

The map F merges agents whose mobility is restricted to $\mathcal{U}_r$ and leaves the other agents invariant. Let $\mathcal{B}$ denote the image of F. For a Markov chain on $\mathcal{A}$ with transition probabilities W_{ij} and stationary distribution $\pi(\cdot)$, we can define a new Markov chain on $\mathcal{B}$ with transitions $\hat{W}_{kl}$

$$\hat{W}_{kl} = \frac{1}{\sum_{i:F(i)=k}\pi(i)} \sum_{i:F(i)=k}\sum_{j:F(j)=l} \pi(i)W_{ij}. \quad (3)$$

This is the *induced chain* from the function F [17, Ch. 4, p. 37]. The stationary distribution of this chain is $\hat{\pi}(k) = \sum_{i:F(i)=k}\pi(i)$.

We can express the relaxation time of a Markov chain in terms of the Dirichlet form [17]. Given a real-valued function g on the state space of the Markov chain with transition matrix W and stationary distribution $\pi(\cdot)$, the Dirichlet form is given by

$$\mathcal{D}(g,g) = \frac{1}{2}\sum_{k,l}\pi(k)W_{kl}(g(k)-g(l))^2.$$

The relaxation time is then given by

$$T_{\text{relax}}(W) = \sup_g \left\{ \frac{\sum_k \pi(k)g(k)^2}{\mathcal{D}(g,g)} : \sum_k \pi(k)g(k) = 0 \right\}. \quad (4)$$

The following contraction principle shows that T_{relax} for an induced chain is at most that of the original chain. This result is claimed in [17, Ch. 4, p. 37], and here we provide a brief proof.

Claim 1: Let $\mathcal{M}$ be a Markov chain on a finite state space $\mathcal{A}$ with transition matrix W and let $F : \mathcal{A} \to \mathcal{B}$ be an arbitrary mapping. Then, the relaxation time of the chain $\hat{\mathcal{M}}$ on $\mathcal{B}$ with transition matrix $\hat{W}$ given by (3) induced by F lower bounds the relaxation time of the original chain

$$T_{\text{relax}}(\hat{W}) \leq T_{\text{relax}}(W). \quad (5)$$

We use the extremal property of the relaxation time in (4). Let $\hat{g}$ achieve the supremum in (4) for the induced chain given by $\hat{W}$. We can create a function g from $\hat{g}$ to lower bound $T_{\text{relax}}(\mathcal{M})$. Let $\mathcal{U}_k = \{i : F(i) = k\}$ for each $k \in \mathcal{B}$. Simply set $g(i) = \hat{g}(k)$ for $i \in \mathcal{U}_k$. Then

$$\sum_{i\in\mathcal{A}}\pi(i)g(i)^2 = \sum_{k\in\mathcal{B}}\hat{\pi}(k)\hat{g}(k)^2.$$

Note that $\{\mathcal{U}_k : k \in \mathcal{B}\}$ forms a disjoint partition of $\mathcal{A}$. For this function g, using (3) yields

$$\begin{aligned}\mathcal{D}(g,g) &= \frac{1}{2}\sum_{i,j\in\mathcal{A}}\pi(i)W_{i,j}(g(i)-g(j))^2 \\ &= \frac{1}{2}\sum_{k,l\in\mathcal{B}}\left(\sum_{i\in\mathcal{U}_k}\sum_{j\in\mathcal{U}_l}\pi(i)W_{ij}\right)(\hat{g}(k)-\hat{g}(l))^2 \\ &= \frac{1}{2}\sum_{k,l\in\mathcal{B}}\hat{\pi}(k)\hat{W}_{kj}\end{aligned}$$

and therefore the Dirichlet form $\mathcal{D}(g,g) = \mathcal{D}(\hat{g},\hat{g})$. Therefore, the supremum of (4) for the original chain is at least as large as that for the induced chain, proving (5). ■

Note that while the mixing time of a Markov chain decreases when states are merged, the same is not true for other quantities like the expected time to go from one state to another. The preceding lemma and Theorem 1 give a lower bound on the benefit on the convergence speed of gossip in a network of mobile nodes. In theory, we could optimize the lower bound over all partitions $\{\mathcal{U}_r\}$, but for our examples there are "obvious" partitions that yield meaningful lower bounds. We turn first to the $\sqrt{n} \times \sqrt{n}$ torus.

Corollary 1 (Torus With Horizontal Mobility): Let $G = (\mathcal{V}, \mathcal{E})$ be the $\sqrt{n} \times \sqrt{n}$ torus and suppose that the set of agents $\mathcal{A} = \mathcal{V}$. Let the mobility pattern for the (i,j)th agent be uniformly distributed on the set $\mathcal{U}_i\{(i,k) : k \leq \sqrt{n}\}$, which corresponds to mobility only in the horizontal direction. Then

$$T_{\text{ave}}(n, \epsilon) = \Omega\left(n^2 \log \epsilon^{-1}\right). \tag{6}$$

Proof: Let $\mathcal{U}_i = \{(i,j) : j = 1, 2, \ldots, \sqrt{n}\}$ be the ith row of the torus, so $\{\mathcal{U}_i\}$ partitions $\mathcal{V}$. Consider two agents, one starting at (i,j) and the other at (k,l), where $k = i \pm 1 \bmod \sqrt{n}$. Then, the probability in the algorithm that (i,j) and (k,l) average with each other is the chance that (i,j) is selected times the probability (over the mobility) that (i,j) and (k,l) are adjacent to each other times the chance that (i,j) selects (k,l) out of its neighbors. We can upper bound this probability

$$W_{ij} = O\left(\frac{1}{n} \times \frac{1}{\sqrt{n}}\right).$$

The chain induced from this partition is a cycle with $\sqrt{n}$ states, where each state corresponds to a row in the original Markov chain. The transitions from row k to row $l = (k \pm 1) \bmod \sqrt{n}$ are given by (3)

$$\begin{aligned}\hat{W}_{kl} &= \frac{1}{\sum_{i:F(i)=k} \pi(i)} \sum_{i:F(i)=k} \sum_{j:F(j)=l} \pi(i) W_{ij} \\ &= \sqrt{n} \cdot \sqrt{n} \cdot \sqrt{n} \cdot \frac{1}{n} \cdot O\left(\frac{1}{n} \times \frac{1}{\sqrt{n}}\right) \\ &= O\left(\frac{1}{n}\right).\end{aligned}$$

Therefore the self-transition for each state is $1 - O(1/n)$. Let $\alpha = \hat{W}_{kl}$, and note that $\hat{W}_{kl}$ is the same for each pair of adjacent rows. The matrix $\hat{W}$ is circulant and generated by the vector $(\alpha, 1 - 2\alpha, \alpha, 0, \ldots, 0)$. The eigenvalues are given by the discrete Fourier transform of the vector (cf. [16])

$$\lambda_k(\hat{W}) = 1 - 2\alpha + 2\alpha \cos\left(\frac{(k-1)2\pi}{\sqrt{n}}\right).$$

In particular, the second-largest eigenvalue can be bounded using the Taylor expansion of the cosine

$$\lambda_2(\hat{W}) \geq 1 - 2\alpha + 2\alpha\left(1 - \frac{1}{2}\frac{4\pi^2}{n}\right) = 1 - O\left(\frac{1}{n^2}\right).$$

Therefore, the relaxation time is

$$T_{\text{relax}} = \Omega(n^2)$$

and the averaging time is bounded by Theorem 1. ∎

The preceding theorem shows that allowing nodes to move in only one direction gives the same order convergence time as the torus without any node mobility. That is, sometimes mobility can yield no significant benefits in terms of convergence. In the case where we add a single agent moving in the vertical direction, we still do not gain anything. The proof follows from the same arguments as Corollary 1.

Corollary 2 (A Single Vertical Mover Does Not Help): Let $G = (\mathcal{V}, \mathcal{E})$ be the $\sqrt{n} \times \sqrt{n}$ torus and suppose that the set of agents $\mathcal{A} = \mathcal{V} \cup \{e\}$. Let the mobility pattern for the (i,j)th agent in $\mathcal{V}$ be uniformly distributed on the set $\{(i,k) : k \leq \sqrt{n}\}$, which corresponds to mobility only in the horizontal direction. Let the mobility pattern for e be uniform on $\{(i,1) : i \in \sqrt{n}\}$. Then, for this gossip algorithm

$$T_{\text{ave}}(n, \epsilon) = \Omega\left(n^2 \log \epsilon^{-1}\right).$$

We could prove in a similar way that adding a constant number of agents in the vertical direction does provide better than constant improvement in the convergence time. In the next section, we use this approach to show that 1-D unidirectional mobility cannot help speed up the convergence time of gossip on RGGs as well. Boyd *et al.* [6] have shown that the averaging time for standard pairwise gossip on the RGG is $\Theta(nr^{-2} \log \epsilon^{-1})$, which for $r(n) = \Theta(\sqrt{n^{-1} \log n})$ is $\Theta((n^2/\log n) \log \epsilon^{-1})$.

C. Upper Bounds

For our upper bounds, we use the canonical path method [19], which we summarize here for completeness. For any ergodic and reversible Markov chain on a state space Ω, for each pair i, j of states define the *capacity* of a directed edge $e = (i,j)$ to be

$$C(e) = \pi(i)\bar{W}_{ij}.$$

For each pair of states, we define a *demand* $D(i,j) = \pi(i)\pi(j)$. A *flow* is any way of routing $D(i,j)$ units of "liquid" from i to j for all pairs i, j simultaneously. Formally, a flow $F : \mathcal{P} \to \mathbb{R}^+$ is a function on the set $\mathcal{P}$ of all simple paths on the transition graph of the Markov chain that satisfies the demand

$$\sum_{p \in \mathcal{P}_{ij}} F(p) = D(i,j)$$

where $\mathcal{P}_{ij}$ denotes all the paths from i to j.

For a flow F, we can define the *load* on an edge e to be total flow routed across that edge

$$f(e) = \sum_{i,j \in \Omega} \sum_{p \in \mathcal{P}_{ij} : e \in p} F(p).$$

The *cost* of a flow F is the maximum overload of any edge

$$\rho(F) = \max_e \frac{f(e)}{C(e)}.$$

Finally, define the *length* of a flow $l(f)$ to be longest flow-carrying path, i.e., the longest p for which $F(p) \neq 0$.

Using these definitions, we can use the following Poincaré inequality [19] to yield an upper bound on the inverse spectral gap (relaxation time) of the Markov chain

$$\frac{1}{1-\lambda_2(\bar{W})} \leq \rho(F)l(F).$$

Intuitively, if there are no "bottlenecks" on the transitions for every pair of states, the relaxation time of the chain will be very small. Any flow F gives an upper bound that depends on the cost $\rho(F)$ of its most congested edge.

Corollary 3 (Full Mobility Is Optimal): Let the area in which the agents move be given by the graph $G = (\mathcal{V}, \mathcal{E})$ corresponding to the $\sqrt{n} \times \sqrt{n}$ discrete lattice on the torus. Let the set of agents $\mathcal{A} = \{1, 2, \ldots, \sqrt{n}\}^2$ with initial locations equal to $\mathcal{V}$. Suppose the mobility pattern of every agent in $\mathcal{A}$ is the uniform distribution on the set of all locations $\mathcal{V}$, which corresponds to full mobility. Then, for this gossip algorithm

$$T_{\text{ave}}(n, \epsilon) = O\left(n \log \epsilon^{-1}\right).$$

Proof: The stationary distribution is uniform, so $\pi(i) = 1/n$ for all i and the demand $D(i,j) = 1/n^2$ for all pairs (i,j). Furthermore, the probability of i and j averaging is $\Omega(1/n^2)$, so the state diagram of the Markov chain is the complete graph with edge capacities $\Omega(1/n^3)$. The simplest flow is to route directly the demand $1/n^2$ on the edge from i to j, which gives a cost of $O(n)$ with a flow of length 1, so the relaxation time is $O(n)$. ■

A slightly less simple example is a cycle with one fully mobile agent. The cycle has averaging time $\Theta(n^3 \log \epsilon^{-1})$ (see [16]). With one mobile agent the averaging time drops to $O(n^2 \log \epsilon^{-1})$.

Corollary 4 (Cycle With One Fully Mobile Agent): Let the area in which the agents move be given by the graph $G = (\mathcal{V}, \mathcal{E})$ corresponding to the cycle of length n and let there be $n+1$ agents $\mathcal{A} = \mathcal{B} \cup \{v'\}$, where $\mathcal{B} = \mathcal{V} = \{1, 2, \ldots n\}$. The initial locations of the agents in $\mathcal{B}$ are the locations of $\mathcal{V}$ and the agents in $\mathcal{B}$ cannot move. The agent v' has mobility uniformly distributed on $\mathcal{V}$ with initial location 1. Then, for this gossip algorithm

$$T_{\text{ave}}(n, \epsilon) = O\left(n^2 \log \epsilon^{-1}\right).$$

Proof: The stationary distribution for this chain is uniform, so $\pi(i) = 1/(n+1)$ for all i in $\mathcal{A}$. The probability that i and j average for $i, j \in \mathcal{V}$ is 0 unless i and j are neighbors. Otherwise, with probability $\frac{3}{n}$ the mobile node v' is a neighbor of i, so

$$P_{ij} = \frac{1}{n}\left(\left(1 - \frac{3}{n}\right) \cdot \frac{1}{2} + \frac{3}{n} \cdot \frac{1}{3}\right) = \frac{1}{2n}\left(1 - \frac{1}{n}\right).$$

For $i \in \mathcal{A}$ and $j = v'$, we have

$$P_{iv'} = \frac{1}{n} \cdot \frac{3}{n} \cdot \frac{1}{3} = \frac{1}{n^2}.$$

Thus, the capacities are

$$C(i,j) = \begin{cases} \frac{1}{2n(n+1)}\left(1 - \frac{1}{n}\right) & j \in \mathcal{V} \\ \frac{1}{n^2(n+1)} & j = v'. \end{cases}$$

The demand is just $D(i,j) = 1/(n+1)^2$ between each pair of nodes.

To construct a flow F, we just route all flow through the mobile agent v'. An edge (i, v') for $i \in \mathcal{B}$ carries n flows to all agents $j \neq i$, each of size $1/(n+1)^2$ for a total of $f(i, v') = n/(n+1)^2$. Similarly, any edge (v', i) carries the same total flow. All flows are of length 2, so $l(F) = 2$. The overload is

$$\rho(F) = \frac{n/(n+1)^2}{1/(n^2(n+1))} = \frac{n^3}{(n+1)}.$$

And thus for large n, we get an upper bound of $O(n^2)$ for the relaxation time of the chain. The averaging time, then, follows from Theorem 1. ■

V. Examples Revisited

We now turn to our examples of mobility and derive scaling results for gossip with mobility. For the torus, we will show that local mobility in a square of area m^2 cuts the convergence time by m^2 and adding m fully mobile agents cuts the convergence time by m. For the RGG, we will prove the same result for bidirectional mobility and a lower bound for unidirectional mobility.

A. Torus

1) Local Mobility: An important step in bridging the mobility model here with more reasonable mobility models is to consider local mobility, in which an agent moves uniformly in a square of side length $(2m+1)$ centered at its initial location.

Theorem 3: Consider gossip with n agents on the $\sqrt{n} \times \sqrt{n}$ torus $\mathcal{G}$. Let the agent initially at a location i have mobility uniform in a square of side-length $2m+1$ centered at i. Then, the averaging time is

$$T_{\text{ave}}(n, \epsilon) = O\left(\frac{n^2 \log m}{m^2} \log \epsilon^{-1}\right).$$

Proof: Divide the grid into squares of side length m. Initially, each square contains m^2 agents. Let a_i refer to the agent whose initial location is i and let $s(a_i)$ refer to the square containing i. The mobility of agent a_i covers $s(a_i)$ and intersects the squares adjacent to it. For each pair of agents, we must route $D(i,j) = 1/n^2$ units of flow. We will do this by routing flows in L-shaped paths, as shown in Figs. 1 and 2. Since a_i's mobility intersects the squares adjacent to $s(a_i)$, there is a nonzero probability that agent a_i will communicate with an agent $a_{i'}$ whose square $s(a_{i'})$ is adjacent to $s(a_i)$.

Assign the $1/(m^2 n^2)$ units of flow to each agent $a_{i'}$ whose initial location is in the square adjacent to $s(a_i)$. There are m^2 such agents. Each agent then routes $1/(m^4 n^2)$ units of flow to each agent $a_{i''}$ in the next square along the L-shaped path. The flow is routed only along edges (j, j') such that $s(a_j)$ and $s(a_{j'})$ are different. Each left-to-right edge carries flow from the $O(\sqrt{n}/m)$ squares to the left of it. These flows are routed to the $O(n/m^2)$ squares to the right and above it for a total of $O(n^{3/2}/m^3)$ pairs (i,j) that are routed through each square. Each square has m^2 agents so there are $O(n^{3/2}/m)$ flows carrying $1/(n^2 m^2)$ per flow, so the load on the edge is

$$f(i,j) = O\left(\frac{1}{\sqrt{n} m^3}\right).$$

The same bound holds for down-to-up edges.

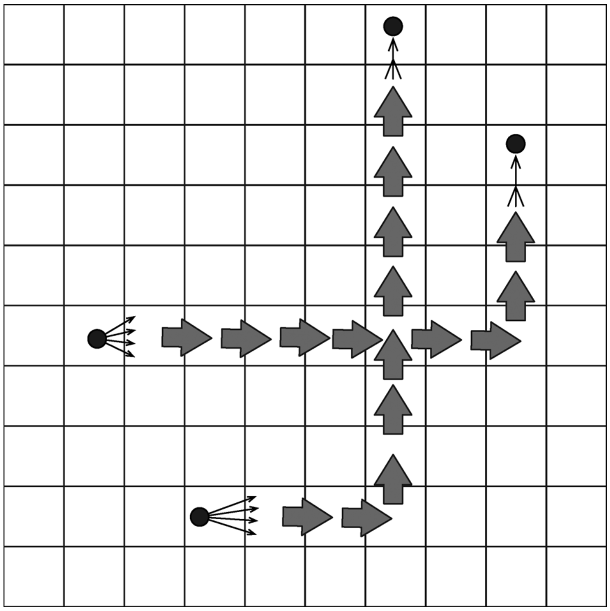

Fig. 1. Routing flow in the local mobility model. Nodes route flows along L-shaped paths through the squares.

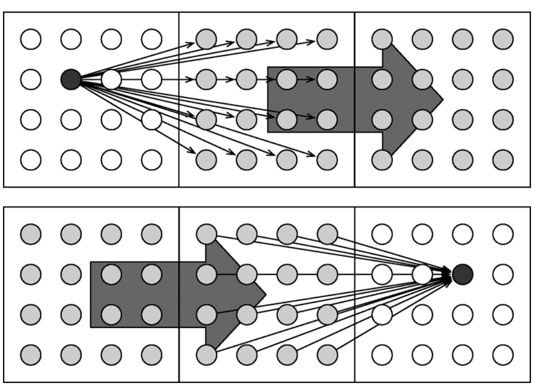

Fig. 2. Routing flow in the local mobility model. As illustrated on top, for node i to send to node j, it evenly divides the flow and sends it to all node in the adjacent square in the L-shaped path. Each node in the adjacent square routes that flow uniformly to every node in the next square in the path. At the end of the route, as illustrated on the bottom, the nodes in the square adjacent to the destination j transmit their received flows directly to j.

To find the capacity of these edges, we calculate the probability that agents i and k in adjacent squares average with each other. The probability is $1/n$ to select agent i and the overlap in agent i and k's mobility area is $\Omega(m^2)$, so the chance i and k are adjacent after moving is $\Omega(1/m^2)$. With high probability, there will be no more than $O(\log m)$ nodes for i to choose from, so the chance of selecting k is at worst $\Omega(1/\log m)$. Thus

$$C(i,k) = \Omega\left(\frac{1}{n^2 m^2 \log m}\right)$$

The maximum length of any flow is $O(\sqrt{n}/m)$, so the Poincaré inequality gives

$$\frac{1}{1-\lambda_2(\bar{W})} = O\left(\frac{n^2 \log m}{m^2}\right).$$

■

2) Adding Mobile Agents: The question motivating this work is this : how much can agent mobility improve the convergence speed of gossip or consensus algorithms? Put another way, how much mobility is needed to gain a certain factor improvement in the convergence? A simple model for which we can answer this question is the following: consider n static agents in the $\sqrt{n} \times \sqrt{n}$ torus together with m mobile agents whose mobility μ_i is uniform on the torus. We use our techniques from earlier sections in the following to show that the averaging time of gossip in this model is $\Theta(n^2/m \log \epsilon^{-1})$, which for $m = n^\alpha$ is $\Theta(n^{2-\alpha})$. For example, adding $\sqrt{n}$ mobile nodes can speed convergence by a factor of $\sqrt{n}$.

Theorem 4: Let the set of locations be given by the $\sqrt{n} \times \sqrt{n}$ discrete lattice on the torus $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Let there be $n+m$ agents $\mathcal{A} = \mathcal{S} \cup \mathcal{M}$ where the n static agents $\mathcal{S}$ are positioned on the n nodes of the torus and do not move. and the m mobile agents $\mathcal{M}$ have mobility that is uniform on $\mathcal{V}$, where $m < n$. Then, the averaging time is given by

$$T_{\text{ave}}(n, \epsilon) = \Theta\left(\frac{n^2}{m} \log \epsilon^{-1}\right).$$

Proof: We first show that for $i \in \mathcal{S}$ and $j \in \mathcal{M}$, the probability P_{ij} that agent i contacts agent j and averages is $\Theta(1/n(m+n))$. Agent i is selected with probability $1/(m+n)$ and agent j is in the neighborhood of agent i with probability $5/n$. Therefore

$$P_{ij} = \frac{5}{n(m+n)} \sum_{l=0}^{m-1} \frac{1}{5+l} \mathbb{P}(L = l)$$

where L is the number of agents in $\mathcal{M}$ that land in the neighborhood of i. The summation is just

$$\sum_{l=0}^{m-1} \frac{1}{5+l} \mathbb{P}(L = l) = \mathbb{E}[1/(5+L)]$$

which is clearly upper bounded by 1, so

$$P_{ij} = O\left(\frac{1}{n(m+n)}\right).$$

Since $1/(5+L)$ is convex, Jensen's inequality can be used to obtain a lower bound

$$\mathbb{E}[1/(5+L)] \geq 1/\mathbb{E}[5+L] = 1/(5+5m/n).$$

Therefore, $P_{ij} = \Omega(1/n(m+n))$. By symmetry, we have the same bound on P_{ji}.

To get the lower bound, consider the function $G : \mathcal{S} \cup \mathcal{M} \to \mathcal{S} \cup \{M\}$ that is the identity on $\mathcal{S}$ and merges $\mathcal{M}$ into a single state M. We can bound the transition probabilities of the new chain using (3)

$$\begin{aligned} \hat{W}_{Mi} &= \frac{1}{\sum_{j \in \mathcal{M}} \pi(j)} \sum_{j \in \mathcal{M}} \pi(j) \frac{P_{ij} + P_{ji}}{2} \\ &= \Theta\left(\frac{1}{n(m+n)}\right) \\ \hat{W}_{iM} &= \frac{1}{\pi(i)} \sum_{j \in \mathcal{M}} \pi(i) \frac{P_{ij} + P_{ji}}{2} \\ &= \Theta\left(\frac{m}{n(m+n)}\right). \end{aligned}$$

For $i, k \in \mathcal{S}$ we have $\hat{W}_{ik} = \bar{W}_{ik}$.

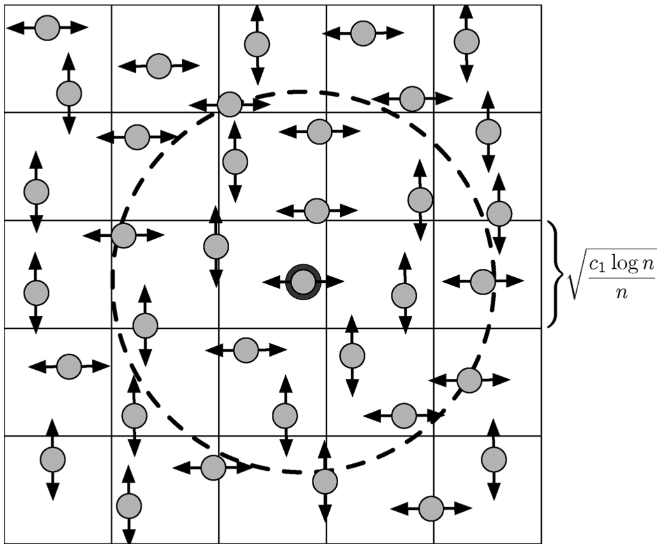

Fig. 3. RGG example with bidirectional 1-D mobility.

The new chain is a torus plus an additional central node M. The probability of transitioning from the torus to the central node is $\Theta((m/n)/(m+n))$ and for transitioning back it is $\Theta((1/n)/(m+n))$. It can be seen (see the Appendix) that the relaxation time for this chain is $\Omega(n^2/m)$ via the extremal characterization in (4). Thus $T_{\text{ave}}(n,\epsilon) = \Omega\left(\frac{n^2}{m}\log\epsilon^{-1}\right)$.

We now turn to the upper bound. As before, we construct a flow on the chain. The demand between any two agents (i,j) is $1/(n+m)^2$. Since $P_{ij} = \Theta(1/n(n+m))$, the capacity

$$C(e) = \Theta(1/n(n+m)^2)$$

for $e = (i,j)$. We must now construct a flow that will yield an upper bound on the relaxation time of n^2/m. For a pair of states $i \in \mathcal{S}$ and $j \in \mathcal{M}$, we assign $1/(n+m)^2$ to the direct path (i,j). For a pair $i \in \mathcal{S}$ and $j \in \mathcal{S}$, we split $1/(n+m)^2$ equally into the m paths (i,k,j) for $k \in \mathcal{M}$. Finally, for $i \in \mathcal{M}$ and $j \in \mathcal{M} \cup \mathcal{S}$ we again route $1/(n+m)^2$ directly on (i,j). Then

$$f((i,j)) = \begin{cases} \frac{1}{(m+n)^2} & i,j \in \mathcal{M} \\ 0 & i,j \in \mathcal{S} \\ \frac{1}{(m+n)^2} + \frac{n}{m}\frac{1}{(m+n)^2} & i \in \mathcal{S},\ j \in \mathcal{S} \cup \mathcal{M}. \end{cases}$$

Therefore, $\rho(F) = \Theta(n^2/m)$. Since all paths are $\Theta(1)$, the Poincaré inequality implies that $T_{\text{relax}}(\bar{W}) = O\left(n^2/m\right)$, so Theorem 1 gives $T_{\text{ave}}(n,\epsilon) = O\left(\frac{n^2}{m}\log\epsilon^{-1}\right)$. ■

B. RGGs

1) Bidirectional Mobility: We now turn to the case where some agents move horizontally and some vertically. We will prove our results for the RGG model, where n nodes are initially placed uniformly in the unit square $\mathcal{G}$. In the bidirectional mobility model, before the gossip algorithm starts, each node flips a fair coin and is assigned to move horizontally or vertically throughout the process. Note that this is a 1-D mobility model since each node is moving only horizontally or vertically throughout the execution of the gossip algorithm, never changing direction (see Fig. 3). Our result is that this mobility model is as good as complete node connectivity.

Theorem 5: Consider the gossip algorithm with n agents under the RGG model and bidirectional mobility. We can choose a connectivity radius $r(n) = \Theta\left(\sqrt{\frac{\log n}{n}}\right)$ such that the gossip averaging time

$$T_{\text{ave}}(n,\epsilon) = \Theta(n\log\epsilon^{-1}).$$

Proof: We start by partitioning the space into a grid of squares of size $c_1\frac{\log n}{n}$. Let B_i denote the number of agents whose initial position was in square i.

It is well known [20], [37]–[40] that a combination of a Chernoff and a union bound yields uniform bounds on the maximum and minimum occupancy of all the squares

$$\mathbb{P}\left(\frac{c_1}{2}\log n \le B_i \le 2c_1\log n\ \forall i\right) \ge 1 - n^{1-c_1/8}\frac{2}{c_1\log n}.$$

By selecting $c_1 \ge 10$, we can show that all the squares have $\Theta(\log n)$ agents with probability at least $1 - \frac{1}{n^2\log n}$, so square occupancies are balanced even if the experiment is repeated n^2 times. We set the transmission radius to $r(n) = \sqrt{5c_1\frac{\log n}{n}}$ to guarantee that an agent in a square can always communicate with any agent in the four adjacent squares.

Recall that, initially, each agent is assigned to be a horizontally moving or vertically moving node by flipping a coin and keeps this directionality throughout the process. Denote by H_i the set of nodes that move horizontally and whose initial position was in the ith row of squares. These agents always stay in the ith row. Similarly, let V_i be the set of agents who move vertically in the ith column of squares.

Each square contains in expectation $c_1\log n$ nodes and there are $\sqrt{\frac{n}{c_1\log n}}$ squares in each row and column. Since each node flips a fair coin and is assigned in a vertically or horizontally moving class, the expected cardinalities will be

$$\mathbb{E}|H_i| = \mathbb{E}|V_i| = \frac{1}{2}c_1\log n\sqrt{\frac{n}{c_1\log n}} = \Theta(\sqrt{n\log n}). \quad (7)$$

Using standard Chernoff bounds, we can show that the cardinalities of $|H_i|, |V_i|$ are sharply concentrated near their expectation.

Theorem 1 shows that the averaging time of the gossip algorithm is bounded by the inverse spectral gap (relaxation time) of the average matrix $\bar{W}$, where the expected matrix $\bar{W} = \mathbb{E}W(s)$ is computed over mobility of the nodes and random selection of which nodes are gossiping.

We now proceed to bound the spectral gap using a canonical flow and we need to select paths for every pair of states for the Markov chain defined by $\bar{W}$. The state space is the set of n agents and $\pi(i) = 1/n$ for each agent i since $\bar{W}$ is doubly stochastic. The *capacities* of the edges will be proportional to the entries of $\bar{W}$ (see (2), where $\bar{W}_{ij}$ is the average of the probabilities P_{ij} and P_{ji}, measuring how often agents i and j are pairwise averaged. For each pair of agents (i,j), we must specify how to satisfy the demand $D(i,j) = n^{-2}$ by assigning flows to some (appropriately chosen) paths in $\mathcal{P}_{ij}$.

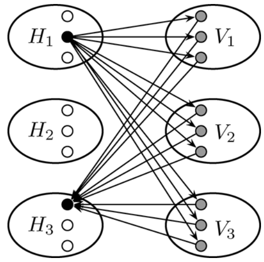

Fig. 4. Routing flow from a node in H_1 to a node in set H_3. The flow is routed from the node in H_1 to all nodes in the sets $\{V_i\}$ and then back to the node in H_3.

Our flow construction uses four different cases depending on whether i and j move horizontally or vertically

1) Case 1: Suppose $i \in H_k$ and $j \in H_l$. To satisfy the demand n^{-2} node i assigns $\Theta(n^{-3})$ units to each path (i, v, j), where $v \in V_r$ for some r. There are $\Theta(n)$ agents who move vertically, so the total flow that reaches j can be made equal to n^{-2} (see Fig. 4).

2) Case 2: Suppose $i \in V_k$ and $j \in V_l$. This is the same as the previous case, except that $\Theta(n^{-3})$ units are assigned to each path (i, h, j) for $h \in H_r$.

3) Case 3: Suppose $i \in H_k$ and $j \in V_l$. To satisfy the demand n^{-2} assign n^{-2} to the direct path (i, j).

4) Case 4: Suppose $i \in V_k$ and $j \in H_l$. We again assign n^{-2} to the direct path (i, j).

Our construction, therefore, only uses the edges in the graph between H sets and V sets. In other words, *it is averaging between nodes that move vertically with nodes that move horizontally* that allows information to spread quickly in the network. The averaging between two nodes in H or V could be omitted and still the bound would not change in order. The total load on an edge $e = (h, v)$ between a horizontal moving agent and a vertical moving agent is the sum of the direct flow (h, v), the sum of the flows (h, v, j) for all horizontal moving i and (i, h, v) for all vertical moving i

$$
\begin{aligned}
f(e) &= \frac{1}{n^2} + \Theta\left(\frac{1}{n^3}\right) \sum |V_r| + \Theta\left(\frac{1}{n^3}\right) \sum |H_r| \\
&= \Theta\left(\frac{1}{n^2}\right).
\end{aligned}
$$

The same bound holds for $e = (v, h)$.

Finally, we calculate the capacity for the edges (v, h). It is sufficient to calculate a lower bound on the probability that agents $v \in V_k$ and $h \in H_l$ average. Agent v is selected with probability $1/n$. Based on our assumptions on the communication radius, v can communicate with $\Theta(\log n)$ neighbors. The probability that v lands in a row within $r(n)$ of row l is $\Theta(\sqrt{n^{-1} \log n})$ and the probability that h lands within $r(n)$ of row k is also $\Theta(\sqrt{n^{-1} \log n})$. Therefore, we have

$$
P_{vh} = \Omega\left(\frac{1}{n}\sqrt{\frac{\log n}{n}}\sqrt{\frac{\log n}{n}}\frac{1}{\log n}\right) = \Omega\left(\frac{1}{n^2}\right).
$$

The capacity of each edge (v, h) is then $C(e) = \Omega(n^{-3})$. By symmetry, the same formulae hold for (h, v).

We can now calculate the overload for this flow on any edge $e = (v, h)$

$$
\frac{f(e)}{C(e)} = \frac{\Theta(n^{-2})}{\Omega(n^{-3})} = O\left(n\right).
$$

Since this holds for all edges, we have $\rho(F) = O(n)$. The maximum length of any path used in the flow is 2, so by the Poincaré inequality we have

$$
T_{\text{relax}}(\bar{W}) = \frac{1}{1 - \lambda_2(\bar{W})} = \rho(F) l(F) = O(n).
$$

Theorem 1 gives the result. ■

One intuition for this result is that bidirectional mobility enables the construction of "short" routes between all pairs of agents. We can derive the identical result for the torus using the same arguments. Under bidirectional mobility, the averaging time for the torus is $O(n \log \epsilon^{-1})$, which is the same as full mobility.

2) Unidirectional Mobility: We now show that unidirectional mobility does not improve the scaling performance for RGGs. This is proved in the same way as the analogous result for the torus.

Corollary 5 (RGG With 1-D Mobility): Consider gossip on the RGG with n agents with the 1-D unidirectional mobility model. Then, for this gossip algorithm

$$
T_{\text{ave}}(n, \epsilon) = \Omega\left(\frac{n^2 \log \epsilon^{-1}}{\log n}\right).
$$

Proof: We first divide the unit square into subsquares of side length $c_1\sqrt{\frac{\log n}{n}}$ for some constant c_1. This creates a $\Theta\left(\sqrt{\frac{n}{\log n}}\right) \times \Theta\left(\sqrt{\frac{n}{\log n}}\right)$ torus on which the mobility can be defined. We must first characterize the Markov chain corresponding to the gossip algorithm under the 1-D unidirectional mobility model. If we set the communication radius to $c_2\sqrt{\frac{\log n}{n}}$, then an agent in the ith row of subsquares can communicate with agents in rows $\{i - c_3, \ldots, i + c_3\}$, where c_3 is again a constant. Moreover, each subsquare will have $\Theta(\log n)$ agents with high probability. Therefore, we can upper bound the probability that an agent in row i will average with an agent in one of the rows $\{i - c_3, \ldots, i - 1, i + 1, \ldots, i + c_3\}$

$$
\beta_{ij} = O\left(\frac{1}{n} \times \sqrt{\frac{\log n}{n}} \times \frac{1}{\log n}\right).
$$

Thus, the chance a given agent averages with someone not in their row of subsquares is $O(1/\sqrt{n^3 \log n})$.

As in the torus, we apply the induced chain method using the partition that merges each row of subsquares. This creates a new Markov chain with $\sqrt{n/\log n}$ states that is a kind of cycle where there are positive transition probabilities from state

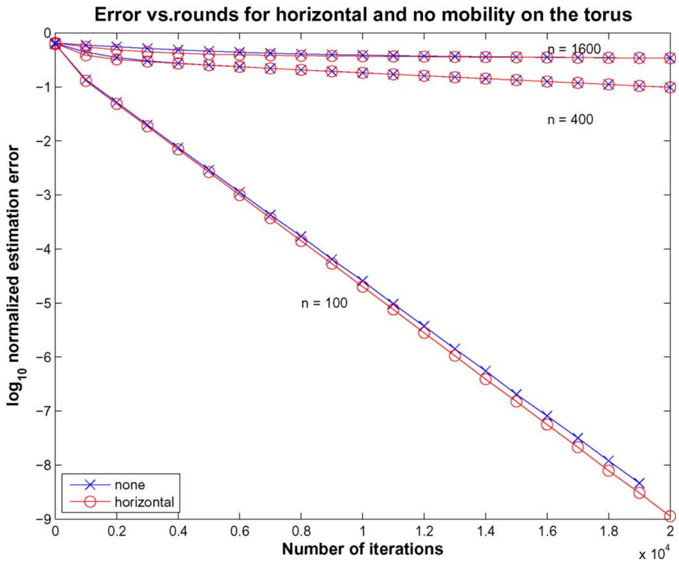

Fig. 5. Log average error versus number of iterations of the gossip algorithm for the torus with no mobility and with horizontal mobility. As the graph size increases, the gap between the two algorithms vanishes.

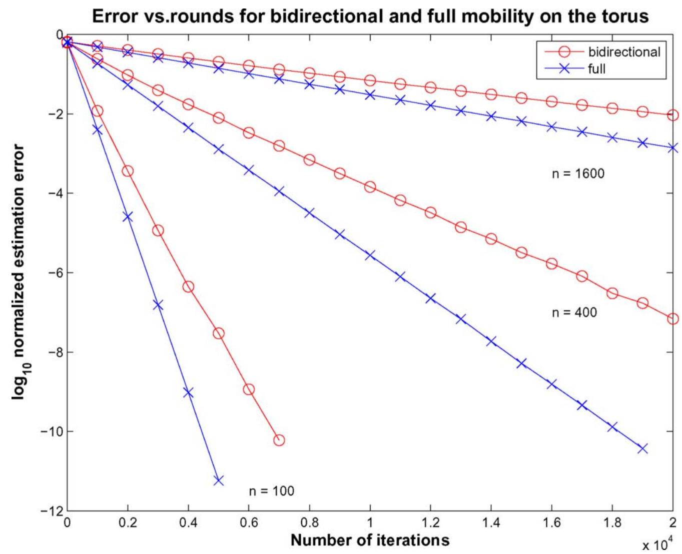

Fig. 6. Log average error versus number of iterations of the gossip algorithm for the torus with full mobility and with bidirectional mobility. As the graph size increases, the gap between the two algorithms shrinks.

k (corresponding to the kth row) to states $l \in \{i-c_3, \ldots, i+c_3\}$. From the analysis of the torus, we can see that from row k to l

$$\begin{aligned}\hat{W}_{kl} &= \frac{1}{\sum_{i:F(i)=k} \pi)(i)} \sum_{i:F(i)=k} \sum_{j:F(j)=l} \pi(i) W_{ij} \\ &= \sqrt{\frac{n}{\log n}} \cdot n \log n \cdot \frac{1}{n} \cdot O\left(\frac{1}{n^{3/2}\sqrt{\log n}}\right) \\ &= O\left(\frac{1}{n}\right).\end{aligned}$$

Let β denote this transition probability. The matrix of this new chain is still circulant and generated by the vector

$$(\beta, \ldots, \beta, 1-2c_3\beta, \beta, \ldots, \beta, 0 \ldots 0).$$

The DFT and Taylor expansion again gives the bound on the second-largest eigenvalue

$$\begin{aligned}\lambda_2(\hat{W}) &= 1 - \beta \cdot O\left(\frac{\log n}{n}\right) \\ &= 1 - O\left(\frac{\log n}{n^2}\right).\end{aligned}$$

Therefore, $T_{\text{relax}}(\hat{W}) = \Omega(n^2/\log n)$. ■

VI. Experiments and Simulations

We can gain some intuition about the benefits of mobility via simulations. All simulations are for a torus with a linearly varying field. Our first main result was a lower bound that shows horizontal mobility is as bad as no mobility in terms of convergence. This is illustrated in Fig. 5, where we can see that, for a range of network sizes, the error under horizontal mobility is close to that of the torus with no mobility. Indeed, as the network size gets larger, the gap vanishes, which suggests that our analysis is tight for this example. Our second major result was a positive one; the bidirectional mobility model was nearly as good as full mobility. This is illustrated in Fig. 6. Although there is a gap between the error decay under the two mobility models, for a fixed error, the number of iterations needed to achieve that error is at most a constant factor more for the bidirectional mobility model.

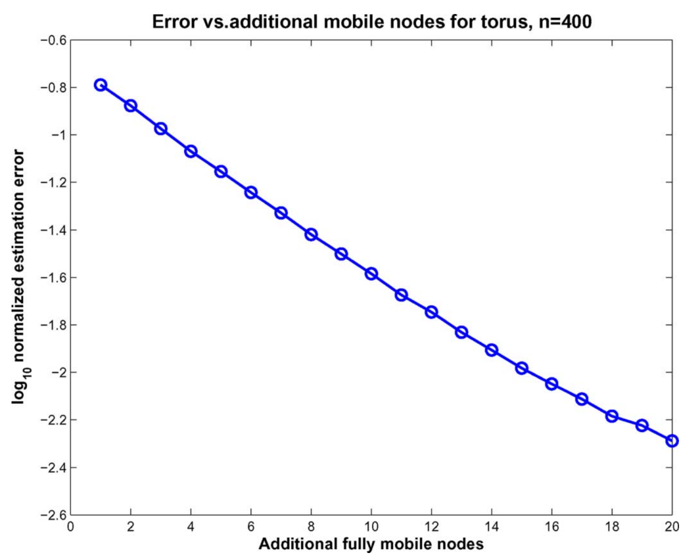

Fig. 7. Adding a few mobile nodes to a static grid can exponentially decrease the estimation error for a fixed number of iterations (20 000).

Our final result was that adding m mobile agents to a static grid with n agents gives a convergence time of $\Theta(n^2/m \log \epsilon^{-1})$. Fig. 7 shows how adding only a few additional mobile agents can dramatically improve the speed of convergence. As we add more nodes, $\log \epsilon$ decreases linearly, which corresponds to an exponential decay in the average error. This suggests that even in large networks, investing in a small number of mobile agents can yield a major benefit in convergence time.

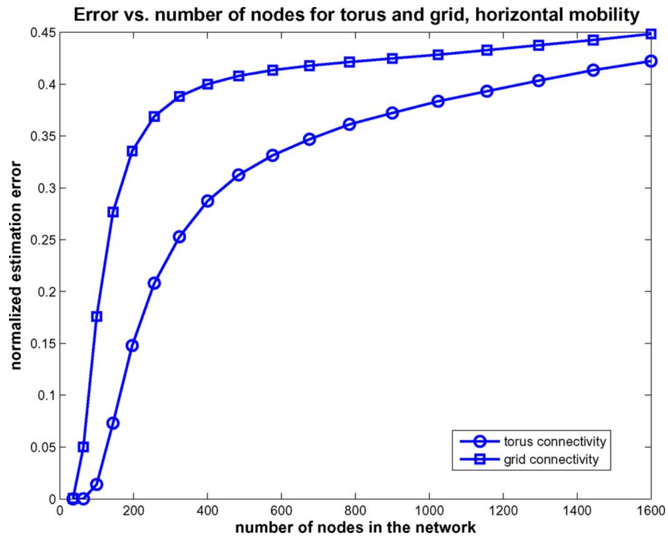

Fig. 8. Gap between the torus and grid versus grid size after 5000 pairwise iterations, averaged over 100 trials, using the uniform horizontal mobility model.

The examples that we consider in this paper are simplifications of real network topologies and real mobility models. It is important to understand how unrealistic these models are. We simulated the difference between the lattice on the torus versus a $\sqrt{n} \times \sqrt{n}$ grid. Fig. 8 shows the error after a fixed number of iterations for increasing grid sizes. Although the algorithm converges faster on the torus, the gap decreases with larger network size. A second question is how the *random walk* mobility model [31], [32] relates to the mobility model in this paper. In order to analyze gossip under such a mobility model, we would need to prove new convergence result for the iterated random matrix products that characterize the evolution of the agents' estimates. It is clear that if each agent moves according to a random walk and the number of steps taken between each gossip iteration is longer than the mixing time of the random walk, then random walk mobility is equivalent to the mobility models considered here. However, for a smaller number of steps, the simulations of the speed of convergence of the algorithm are inconclusive, as there appears to be a dependence on the initial configuration of agents' values. We leave as an open question how to bound the performance Markov random walk models

VII. Discussion and Future Directions

In this paper, we investigated how agent mobility impacts the convergence speed of distributed averaging algorithms by developing new analytical tools derived from the theory of Markov chains. Using these tools, we could show that different mobility patterns can have dramatically different effects depending on the overlap of the mobility paths. Perhaps surprisingly, even a sublinear number of mobile nodes can change the order of gossip messages required for convergence. We note that "mobility" in our model is a variety of time-varying network topology that in practical implementations need not come from the physical mobility of the agents, but can be induced by structured variations in the topology.

The class of mobility models that are amenable to our analysis makes a strong assumption on the speed of the mobility or delay tolerance of the gossip algorithm. One interesting direction for future research involves understanding more realistic mobility models. General mobility models based on Markov chains could be analytically tractable since they would integrate naturally with the Markov structure of the averaging process. Proving that these systems reach a consensus could follow from more general results about the corresponding stochastic process [41]. We conjecture that random walk models with slower mixing times will yield smaller benefits, and that our independent (fast mixing) model is an upper bound. For these models, modifying the pairwise gossip paradigm (cf. [20]) may yield a greater benefit than relying on mobility alone. The impact of node mobility on distributed optimization and general message-passing algorithms on probabilistic graphical models would also be a very interesting research direction.

Another interesting direction is understanding the impact of mobility for more general message-passing algorithms such as distributed convex optimization. The analysis of [42] obtains a convergence theorem similar to the spectral gap and it would be interesting to investigate the scaling behavior of the number of required iterations for the min-sum algorithm to optimize a convex function under out node mobility models.

Appendix

We will construct a g in (4) to show that the mixing time of a torus plus an additional central node M with transition probabilities $\Theta((m/n)/(m+n))$ to M and $\Theta((1/n)/(m+n))$ away from M along with transitions $\Theta(1/n)$ between neighbors in the torus has relaxation time $\Omega(n^2/m)$, where $m < n$. The stationary distribution for this chain has probability $\pi(i) = \Theta(1/(m+n))$ on the nodes $i = 1, 2, \ldots, n$ of the torus and $\pi(M) = \Theta(m/(m+n))$ on M. Let $g(M) = 0$ and g be constant on each column of the torus with the values on the columns being $\{-\alpha, -\alpha+1, \ldots 0, 1, 2, \ldots, \alpha, \alpha, \alpha-1, \ldots, -\alpha+1, -\alpha\}$ for $\sqrt{n}$ even and $\{-\alpha, -\alpha+1, \ldots, \alpha, 0, \alpha, \alpha-1, \ldots, -\alpha\}$ for $\sqrt{n}$ odd, where $\alpha = \Theta(\sqrt{n})$. Then clearly $\sum \pi(i)g(i) = 0$. We can calculate the numerator and denominator in (4)

$$\begin{aligned}\sum_k \pi(k)g(k)^2 &= \frac{1}{m+n}\sqrt{n}4\sum_{i=0}^{\alpha} i^2 \\ &= \Theta\left(\frac{n^2}{m+n}\right) \\ \mathcal{D}(g,g) &= \frac{m/n}{(m+n)^2}\sqrt{n}4\sum_{i=1}^{\alpha} i^2 + \frac{1/n}{m+n}\sqrt{n}4\sum_{i=1}^{\alpha} 1 \\ &= \Theta\left(\frac{1+\frac{mn}{m+n}}{m+n}\right).\end{aligned}$$

Dividing gives the result.

References

[1] A. G. Dimakis, S. Kar, J. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.
[2] J. Tsitsiklis, "Problems in Decentralized Decision-Making and Computation," Ph.D., Dept. Elect. Eng. Comput. Sci., MIT, Cambridge, 1984.
[3] F. Fagnani and S. Zampieri, "Randomized consensus algorithms over large scale networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 4, pp. 634–649, May 2008.

[4] R. Olfati-Saber, J. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.
[5] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. 44th Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2003, pp. 482–491.
[6] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.
[7] D. Spanos, R. Olfati-Saber, and R. Murray, "Approximate distributed Kalman filtering in sensor networks with quantifiable performance," in *Proc. 4th Int. Symp. Inf. Process. Sens. Netw.*, 2005, pp. 133–139.
[8] L. Xiao, S. Boyd, and S. Lall, "A scheme for asynchronous distributed sensor fusion based on average consensus," presented at the presented at the 4th Int. Symp. Inf. Process. Sens. Netw., 2005.
[9] V. Saligrama, M. Alanyali, and O. Savas, "Distributed detection in sensor networks with packet losses and finite capacity links," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4118–4132, Nov. 2006.
[10] B. Nazer, A. Dimakis, and M. Gastpar, "Local interference can accelerate gossip algorithms," in *Proc. 46th Allerton Conf. Commun., Control, Comput.*, 2008, pp. 591–598.
[11] M. Rabbat, J. Haupt, A. Singh, and R. Nowak, "Decentralized compression and predistribution via randomized gossiping," in *Proc. 5th Int. Symp. Inf. Process. Sens. Netw.*, Apr. 2006, pp. 51–59.
[12] D. Mosk-Aoyama and D. Shah, "Computing separable functions via gossip," presented at the presented at the 25th Annu. ACM SIGACT-SIGOPS Symp. Principles Distrib. Comput., Denver, CO, Jul. 2006.
[13] S. Sanghavi, B. Hajek, and L. Massoulie, "Gossiping with multiple messages," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4640–4654, Dec. 2007.
[14] T. C. Aysal, M. E. Yildiz, A. D. Sarwate, and A. Scaglione, "Broadcast gossip algorithms for consensus," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2748–2761, Jul. 2009.
[15] D. Üstebay, B. Oreshkin, M. Coates, and M. Rabbat, "Greedy gossip with eavesdropping," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3765–3776, Jul. 2010.
[16] A. Dimakis, A. Sarwate, and M. Wainwright, "Geographic gossip: Efficient averaging for sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1205–1216, Mar. 2008.
[17] D. Aldous and W. Fill, Reversible Markov Chains and Random Walks on Graphs 2002 [Online]. Available: http://www.stat.berkeley.edu/ aldous/RWG/book.html
[18] P. Diaconis and D. Stroock, "Geometric bounds for eigenvalues of Markov chains," *Ann. Appl. Probabil.*, vol. 1, pp. 36–61, 1991.
[19] A. Sinclair, "Improved bounds for mixing rates of Markov chains and multicommodity flow," in *Combinatorics, Probability and Computing*. Cambridge, U.K.: Cambridge Univ. Press, 1992, vol. 1.
[20] F. Benezit, A. G. Dimakis, P. Thiran, and M. Vetterli, "Order-optimal consensus through randomized path averaging," presented at the presented at the 45th Allerton Conf. Commun., Control, Comput., Urbana-Champaign, IL, Sep. 2007.
[21] D. Bertsekas and J. Tsitsiklis*, Parallel and Distributed Computation: Numerical Methods*. Belmont, MA: Athena Scientific, 1997.
[22] A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, 2007.
[23] T. C. Aysal, M. J. Coates, and M. G. Rabbat, "Distributed average consensus with dithered quantization," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4905–4918, Oct. 2008.
[24] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. Tsitsiklis, "On distributed averaging algorithms and quantization effects," *IEEE Trans. Automat. Control*, vol. 54, no. 11, pp. 2506–2517, Nov. 2009.
[25] R. Carli, F. Bullo, and S. Zampieri, "Quantized average consensus via dynamic coding/decoding schemes," *Int. J. Robust Nonlinear Control*, vol. 20, no. 2, pp. 156–175, 2010.
[26] R. Carli, F. Fagnani, P. Frasca, and S. Zampieri, "Gossip consensus algorithms via quantized communication," *Automatica*, vol. 46, no. 1, pp. 70–80, 2010.
[27] S. Kar and J. Moura, "Distributed consensus algorithms in sensor networks: Quantized data and random link failures," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1383–1400, Mar. 2010.
[28] J. Lavaei and R. M. Murray, "Quantized consensus by means of gossip algorithm," *IEEE Trans. Automat. Control*, vol. 57, no. 1, pp. 19–32, Jan. 2012, to be published.
[29] M. Zhu and S. Martínez, "On the convergence time of asynchronous distributed quantized averaging algorithms," *IEEE Trans. Automat. Control*, vol. 56, no. 2, pp. 386–390, Feb. 2011.
[30] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 477–486, Aug. 2002.
[31] M. Grossglauser and M. Vetterli, "Locating mobile nodes with EASE: Learning efficient routes from encounter histories alone," *IEEE/ACM Trans. Netw.*, vol. 14, no. 3, pp. 457–469, Jun. 2006.
[32] A. Jindal and K. Psounis, "Fundamental mobility properties for realistic performance analysis of intermittently connected mobile networks," in *Proc. 5th Annu. IEEE Int. Conf. Perv. Comput. Commun. Workshop*, 2007, pp. 59–64.
[33] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. 4th Annu. ACM/IEEE Int. Conf. Mobile Comput. Netw.*, 1998, pp. 85–97.
[34] C. Bettstetter, "Mobility modeling in wireless networks: Categorization, smooth movement, and border effects," *ACM Mobile Comput. Commun. Rev.*, vol. 5, pp. 55–66, 2001.
[35] T. Spyropoulos, A. Jindal, and K. Psounis, "An analytical study of fundamental mobility properties for encounter-based protocols," *Int. J. Auton. Adapt. Commun. Syst.*, vol. 1, pp. 215–233, Jul. 2008.
[36] P. Denantes, F. Bénézit, P. Thiran, and M. Vetterli, "Which distributed averaging algorithm should I choose for my sensor network?," in *Proc. 27th Conf. Comput. Commun.*, 2008, pp. 986–994.
[37] M. Penrose*, Random Geometric Graphs*. Oxford, U.K.: Oxford Univ. Press, 2003, Oxford studies in probability.
[38] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.
[39] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Throughput-delay trade-off in wireless networks," in *Proc. 23rd Conf. Comput. Commun.*, 2004, pp. 464–475.
[40] C. Avin and G. Ercal, "On the cover time and mixing time of random geometric graphs," *Theor. Comput. Sci.*, vol. 380, pp. 2–22, Jul. 2007.
[41] B. Touri and A. Nedic, "On ergodicity, infinite flow and consensus in random models," *IEEE Trans. Automat. Control*, vol. 56, no. 7, pp. 1593–1605, Jul. 2011.
[42] C. Moallemi and B. V. Roy, "Convergence of the min-sum algorithm for convex optimization," presented at the presented at the 45th Annu. Allerton Conf. Commun., Control Comput., Monticello, IL, Sep. 2007.

Anand Dilip Sarwate (S'99–M'09) received B.S. degrees in electrical engineering and computer science and mathematics from the Massachusetts Institute of Technology (MIT), Cambridge, in 2002 and the M.S. and Ph.D. degrees in electrical engineering in 2005 and 2008, respectively, from the University of California, Berkeley. From 2008–2011 he was a postdoctoral researcher at the Information Theory and Applications Center at the University of California, San Diego.

Since October 2011 he is a Research Assistant Professor at the Toyota Technological Institute at Chicago. His research interests include information theory, distributed signal processing, machine learning, and privacy.

Dr. Sarwate received the Laya and Jerome B. Wiesner Student Art Award from MIT, and the Samuel Silver Memorial Scholarship Award and Demetri Angelakos Memorial Achievement Award from the EECS Department at University of California at Berkeley. He was awarded an NDSEG Fellowship from 2002 to 2005. He is a member of Phi Beta Kappa and Eta Kappa Nu.

Alexandros G. Dimakis (M'09) received the Diploma degree in electrical and computer engineering from the National Technical University of Athens, Athens, Greece, in 2003 and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of California Berkeley, in 2005 and 2008, respectively.

Currently, he is an Assistant Professor at the Viterbi School of Engineering, University of Southern California, Los Angeles. He has been a faculty member in the Department of Electrical Engineering-Systems since 2009. He was a Postdoctoral Scholar at the Center for the Mathematics of Information (CMI), California Institute of Technology (Caltech), Pasadena, in 2008. His research interests include coding theory, signal processing, and networking, with a current focus on distributed storage, network coding, probabilistic inference, and message passing algorithms.

Dr. Dimakis received the NSF CAREER Award in 2011, the Eli Jury Award in 2008 for his thesis work on codes for distributed storage, and two outstanding paper awards.