Reaching consensus in wireless networks with probabilistic broadcast

Tuncer C. Aysal
Pixsta Research
9 Thorpe Close
Portobello Road
London W10 5XL
Email: can.aysal@gmail.com

Anand D. Sarwate
Information Theory and Applications Center
University of California, San Diego
La Jolla, CA 92093-0447
Email: asarwate@ucsd.edu

Alexandros G. Dimakis
Dept. of Electrical Engineering - Systems
University of Southern California
Los Angeles, CA
Email: dimakis@usc.edu

Abstract—**Reaching consensus in a network is an important problem in control, estimation, and resource allocation. While many algorithms focus on computing the exact average of the initial values in the network, in some cases it is more important for nodes to reach a consensus quickly. In a distributed system establishing two-way communication may also be difficult or unreliable. In this paper, the effect of the wireless medium on simple consensus protocol is explored. In a wireless environment, a node's transmission is a broadcast to all nodes which can hear it, and due to signal propagation effects, the neighborhood size may change with time. A class of non-sum preserving algorithms involving unidirectional broadcasting is extended to a time-varying connection model. This algorithm converges almost surely and its expected consensus value is the true average. A simple bound is given on the convergence time.**

I. INTRODUCTION

Reaching consensus is an important building block underlying more complex protocols in distributed networked systems. In particular, consensus is critical for synchronization, control, data fusion, and load balancing. Randomized consensus algorithms have received significant attention in recent years, motivated by emerging sensor network applications. In this paper we study the effect of physical layer signal propagation effects on the convergence speed of an asynchronous broadcasting "gossip" algorithm for achieving consensus. Consensus algorithms perform iterative updates of a state estimate until the network achieves consensus. The central feature of gossip algorithms is the asynchronous time model in which each update only involves a subset of the sensors (often only two). The nodes in the networks involved in each update are usually neighbors, and in each update the nodes update their state via a simple linear combination of their values. From their introduction in Tsitsiklis's thesis [1], gossip and consensus algorithms have been investigated for several applications in distributed control, sensor networks and distributed signal processing.

One issue with pairwise averaging is that local communication with a small number of nodes may be inefficient. For example, the communication complexity of the randomized gossip algorithm of [2] (measured by number of radio transmissions to drive the estimation error to within $\Theta(N^{-\alpha})$, for any $\alpha > 0$) on the order of $\Theta(N^2 \log N)$ for random

geometric graphs. The proposed "geographic gossip" algorithm combines gossip with geographic routing to improve the convergence rate of random gossiping [3]. Similar to the standard gossip algorithm, a node randomly wakes up, chooses a node randomly in the whole network, rather than in its neighborhood, and performs a pairwise averaging with this node. Geographic gossiping increases the diversity of every pairwise averaging operation. The authors show that the communication complexity is in the order of $O(N^{3/2}\sqrt{\log N})$, which is an improvement with respect to the standard gossiping algorithm. More recently, a variety of the algorithm that "averages along the way" has been shown to converge in $O(N \log N)$ transmissions [4]. Nedic et. al [5] presented constrained consensus algorithms where the estimate of each agent is restricted to lie in a different constraint set. The problem of reaching a consensus and rate of convergence in the values of a distributed system with time-varying connectivity in the presence of delays is studied in [6].

A second issue with pairwise averaging is that a two-way link must be established, making these protocol vulnerable to packet collisions. To overcome the drawbacks of the standard packet based gossip algorithms, a broadcast based gossiping algorithm for wireless sensor networks was also recently studied [7]. In that work a node in the network wakes up uniformly at random according to the asynchronous time model and broadcasts its value. This value is successfully received by the nodes in the predefined radius of the broadcasting node, *i.e.*, connectivity radius. The nodes that have received the broadcasted value update their own state value and the remaining nodes sustain their value. It is shown through simulations that the broadcast gossip achieves a consensus much faster than both random and geographic gossip algorithms, especially for moderate network sizes.

We are interested in the benefits that can be obtained from the broadcast nature of the wireless medium on the convergence of distributed message-passing algorithms. One approach to using the wireless medium in consensus is to enable physical-layer cooperation [8], [9], but this generally requires information-theoretic assumptions involving long vectors of numbers to be averaged. Related recent work uses the wireless medium to eavesdrop other nodes values [10], or to

deal with noisy links and quantization effects [11]–[13]. In this paper we only exploit the broadcast benefit of the physical medium without worrying about these extra complications; they remain to be investigated in future work.

In particular, we are interested in how fading can eventually enable opportunistic longer-range transmissions. Depending on the fading statistics, it may be possible for a the broadcast message of one node to reach other nodes far away in the network. Enforcing a multihop or local connection model as in [3], [4], [10] corresponds to the assumption that long-distance connections are wasteful in terms of power or that the attenuation of the signal due to long distances.

We analyze a probabilistic version of the broadcast-based gossip algorithm originally introduced in [7] and determine analytical convergence results. In particular, we show that it converges almost surely and that its expected consensus value is the true average. The actual consensus value may vary about the true mean, which is a limitation of this and other non-sum-preserving algorithms. We defer this analysis for the full version of this paper, but the techniques of [7] apply directly to this model as well. Even though the consensus value may be different from the true mean, we provide a simple bound on the time to reach consensus. All our results hold for general graphs and fading models. We show that for the special case of power-law decaying probability of successful communication, the convergence rate may be surprisingly fast.

II. PRELIMINARIES AND SHOUT GOSSIP ALGORITHM

In the following, we briefly discuss the graph and time models adopted in this paper. Then, we describe briefly the distributed average consensus problem along with the proposed consensus algorithm.

A. Notation

We will denote the all 1's vector by $\mathbf{1}$ and $J = \frac{1}{N}\mathbf{1}\mathbf{1}^T$. The symbols $\mathsf{P}(\cdot)$ and $\mathsf{E}[\cdot]$ denote probability and expectation. We write $\mathbf{e}_i$ for the i-th elementary vector, and for a subset $K \subset [N]$ we write $\mathbf{e}_K = \sum_{k \in K} \mathbf{e}_k$ for the vector with 1's in the coordinates K and 0's elsewhere. We write $K^c = [N] \setminus K$ for the complement of K in $[N]$. We write P for the matrix (P_{ij}), and $\mathbf{p}_i$ for the i-th row or column of P (note that by definition P is symmetric). We will often think of P as the adjacency matrix of a weighted graph. The Laplacian $L(P)$ for P is

$$L(P) = \operatorname{diag}(P\mathbf{1}) - P . \qquad (1)$$

B. Time Model

We use the asynchronous time model, which is well–matched to the distributed nature of sensor networks [2], [14]. In this model, each sensor node is assumed to have a clock which ticks independently according to a rate μ Poisson process. Consequently, the inter-tick times are exponentially distributed and independent across nodes and over time. This process is equivalent to a single clock whose ticking times form a Poisson process of rate $N\mu$. Let Z_t be the the arrival times of this global process. In expectation, there are

approximately $N\mu$ clock ticks per unit of absolute time but we will always measure time in number of ticks of this (virtual) global clock. We therefore think of time as discretized with the interval $[Z_t; Z_{t+1})$ corresponds to the t–th timeslot. We can adjust time units relative to the communication time so that only one broadcast event occurs in the network at each time slot with high probability.

C. Graph Model

We model our wireless sensor network as a graph G with N vertices or nodes distributed in the plane at locations $\{R_i : i \in [N]\}$ in $\mathbb{R}^2$. The N–node topology of G at time step t is represented by the $N \times N$ distance matrix D, where for $i \neq j$, D_{ij} is the distance between nodes i and j. For example, we may take G to be the random geometric graph, where the N sensor locations are chosen uniformly and independently in a unit square area.

D. Average Consensus

At time slot $t \geq 0$, each node $i = 1, 2, \ldots, N$ has an estimate $x_i(t)$ of the global average, and we use $x(t)$ to denote the N-vector of these estimates. The ultimate goal is to use the minimal amount of communication to drive the estimate $x(t)$ as close as possible to the average vector $\bar{x}(0)\mathbf{1}$, where $\mathbf{1}$ is the vector of all 1's and

$$\bar{x}(0) = \frac{1}{N} \sum_{i=1}^{N} x_i(0). \qquad (2)$$

Because our algorithms are randomized, the quantity $x(t)$ for $t > 0$ is a random vector even though we assume $x(0)$ is deterministic.

E. Broadcast consensus protocol

Suppose at time step t, node $i \in \mathcal{N} = \{1, 2, \ldots, N\}$ clock ticks. Then node i activates and the following events occur in the network:

1) Node i shouts/broadcasts its current state value, $x_i(t)$ over the wireless medium.
2) The shouted value is successfully received by the node j in the network with probability P_{ij} that is a monotonically decreasing function $f(\cdot)$ of the distance D_{ij} between nodes i and j.
3) Let $\mathcal{J}$ denote the set of nodes that successfully received the shouted state value $x_i(t)$. Each node $j \in \mathcal{J}$ updates its own state value according to

$$x_j(t+1) = \gamma x_j(t) + (1-\gamma)x_i(t), \; j \in \mathcal{J} \qquad (3)$$

where $\gamma \in (0, 1)$ is the mixing parameter of the algorithm.
4) The remaining nodes in the network, *i.e.*, the nodes that did not successfully receive the shouted value, including i, update their state value as

$$x_j(t+1) = x_j(t), \; j \in \mathcal{N} - \mathcal{J}. \qquad (4)$$

F. Impact of different fading models

One example for the reception probability P_{ij} is to assume a lognormal shadowing [15] so that the signal-to-noise (SNR) at j from a transmission by i is given by

$$\log \mathsf{SNR} = A - \alpha \log D_{ij} + S \, , \qquad (5)$$

where α is the pathloss and S is the shadowing effect. The probability for a node to be above a given SNR threshold (indicating successful reception) can serve as a model for P_{ij}.

A simpler model is to set P_{ij} inversely proportional to a power of the distance D_{ij} and directly proportional to the transmit power $\mathcal{P}$:

$$P_{ij} \propto \frac{\mathcal{P}}{D_{ij}^{\beta}} \, . \qquad (6)$$

In this simplified model, for a network of N nodes with constant density, the reception probability for the two most distant nodes scales like $N^{-\beta/2}$. For a network with constant density, the transmit power is typically scaled down to save power, which would give a more complicated dependence on N.

G. Matrix updates

Let $\mathcal{A}(t)$ denote the random index of the node of which the internal clock ticked. Formally, let $x(t)$ denote the vector of state values at the end of time-slot t. Then, the network-wide update is given by

$$x(t+1) = W(t)x(t) \qquad (7)$$

where the random matrix $W(t)$ is characterized as following:

$$\mathsf{P}(W_{[j]}(t) = \gamma e_j + (1-\gamma)e_i | \mathcal{A}(t) = i) = P_{ij} \qquad (8)$$

and

$$\mathsf{P}(W_{[j]}(t) = e_j | \mathcal{A}(t) = i) = 1 - P_{ij} \qquad (9)$$

where $P_{ii} = 0$, $W_{[j]}$ denotes the j–th row of the matrix W and e_j denotes the row vector of zeros only which the j–th element is one.

III. ANALYSIS OF THE ALGORITHM

We now turn to the analysis of the algorithm. In subsection III-A we provide some properties of the averaging matrices. In III-B we prove that the algorithm converges almost surely, and in III-C that in expectation it converges to the true average. In III-D we provide a simple bound on the convergence time of the algorithm in terms of the matrix P.

A. Averaging Matrix Properties

The following results reveal important properties regarding the random averaging matrices $\{W(t) : t \geq 0\}$.

Lemma 1 *The random averaging matrix $W(t)$ obeys*

$$\mathsf{P}(W(t)\mathbf{1} = \mathbf{1}) = 1 \qquad (10)$$

and

$$\mathsf{P}\left(\mathbf{1}^T W(t) = \mathbf{1}^T\right) = N^{-1} \sum_{i=1}^{N} \prod_{j=1, j \neq i}^{N} (1 - P_{ij}) \qquad (11)$$

for all $t \geq 0$.

Proof: See the Appendix. ∎

The above Lemma reveals two important properties of the gossip algorithm: 1) $c\mathbf{1}$ for some $c \in \mathbb{R}$ is a fixed point of the gossip algorithm. If the algorithm converges to a consensus, the network will not leave the consensus state; 2) However, it also shows that the sum (and therefore the average) of the vector of node values is very unlikely to be preserved at each step. In fact, the sum is preserved only when none of the nodes in the network receives the value.

The next lemma describes the expected values of various functions of the weight matrices $W(t)$ which will be useful in deriving eigenvalue bounds and convergence results for the algorithm.

Lemma 2 *We have the following:*

$$W \triangleq \mathsf{E}[W(t)] = I - \frac{1-\gamma}{N} L(P) \qquad (12)$$

$$W' \triangleq \mathsf{E}[W(t)^T W(t)] = I - \frac{2\gamma(1-\gamma)}{N} L(P) \qquad (13)$$

$$\begin{aligned}
W'' \triangleq \mathsf{E}[W(t)^T J W(t)] = J &+ \frac{(1-\gamma)^2}{N^2} L(P)^2 \\
&+ \frac{2(1-\gamma)^2}{N^2} L(P) \\
&- \frac{2(1-\gamma)^2}{N^2} L(P \odot P) \, , \quad (14)
\end{aligned}$$

where $\odot$ is the Hadamard (element by element) product.

Proof: See the Appendix. ∎

In many gossip algorithms, the convergence time is related to the relaxation time of a certain Markov chain associated to the algorithm. In our case the time to consensus is related to the largest eigenvalue of the matrix $\lambda_1(\mathsf{E}[W(t)^T(I-J)W(t)])$, and we can derive a bound using the Poincaré inequality [16], [17].

Lemma 3 (Simple eigenvalue bound) *We have*

$$\frac{1}{1 - \lambda_1(\mathsf{E}[W(t)^T(I-J)W(t)])} \leq \frac{2\gamma(1-\gamma)}{\min\{P_{ij}\}} \qquad (15)$$

Proof: We must find the largest eigenvalue of $W' - W''$:

$$\begin{aligned}
W' - W'' = I - J &- \frac{2\gamma(1-\gamma)}{N} L(P) - \frac{(1-\gamma)^2}{N^2} L(P)^2 \\
&- \frac{2(1-\gamma)^2}{N^2} L(P) + \frac{2(1-\gamma)^2}{N^2} L(P \odot P)
\end{aligned}$$
$$(16)$$

Note that $\mathbf{1}$ is an eigenvector for each term in (16), and in particular that $\mathbf{1}$ corresponds to the only non-zero eigenvalue

of J. Therefore the eigenvectors of $W' - W''$ are equal to 0 and the eigenvalues of

$$I - \frac{2\gamma(1-\gamma)}{N}L(P) - \frac{(1-\gamma)^2}{N^2}L(P)^2 - \frac{2(1-\gamma)^2}{N^2}L(P)$$
$$+ \frac{2(1-\gamma)^2}{N^2}L(P \odot P) \ . \tag{17}$$

The spectral radius of the Laplacian matrix $L(P)$ is bounded. In particular, the eigenvalues of the Laplacian matrix are bounded by $\max\{|\lambda_k(L(P))|\} \leq 2N$ [18]. First, we can combine the last two terms to get $-\frac{2(1-\gamma)^2}{N^2}L(P - P \odot P)$. Since all entries of P are upper bounded by 1, the matrix $P - P \odot P$ has all positive entries. Since we are interested in an upper bound on the second largest eigenvalue of the matrix in (17), we will ignore the negative terms corresponding to $L(P - P \odot P)$ and $L(P)^2$.

We therefore let $\mu = 2\gamma(1-\gamma)$ and focus on the second largest eigenvalue of the matrix

$$Q = I - \frac{\mu}{N}L(P) \ . \tag{18}$$

This matrix corresponds to a Markov chain with the uniform stationary distribution, and thus we are interested in bounding the mixing time of this Markov chain. In particular, we would like to lower bound the spectral gap $1 - \lambda_2(Q)$.

In order to bound the spectral gap, we will use the Poincaré inequality [16] and method of canonical paths [17] which has been useful in other gossip algorithm analyses [4], [19]. For each pair i, j of states the *capacity* of a directed edge $e_{i,j}$ is

$$C(e) = \pi(i)\bar{Q}_{ij} \ . \tag{19}$$

We also define a *demand* $D(i, j) = \pi(i)\pi(j)$ for each pair of nodes.

A *flow* F is method of simultaneously routing all the demands $\{D(i, j)\}$ from i to j for all pairs i, j. That is $F : \mathcal{P} \rightarrow \mathbb{R}^+$ is a function on the set $\mathcal{P}$ of all simple paths on the transition graph corresponding to Q that satisfies the demand:

$$\sum_{p \in \mathcal{P}_{ij}} F(p) = D(i, j) \ , \tag{20}$$

where $\mathcal{P}_{ij}$ denotes all the paths from i to j. The length $\ell(\cdot)$ of F is the longest path p for which $F(p) \neq 0$.

The inequality hinges on finding a flow that does not cause route too much more on each edge than its capacity. The *load* induced by F on an edge e is total flow routed across that edge:

$$f(e) = \sum_{p \in \mathcal{P}_{ij} : e \in p} F(p) \tag{21}$$

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

$$\rho(F) = \max_e \frac{f(e)}{C(e)} \ , \tag{22}$$

The Poincaré inequality [17] gives an upper bound on the inverse spectral gap of Q:

$$\frac{1}{1 - \lambda_2(Q)} \leq \rho(F)\ell(F). \tag{23}$$

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

We can construct a trivial flow for our algorithm to get the desired bound. The demand on each edge (i, j) is $1/N^2$, so we can simply route $1/N^2$ on the direct edge (i, j). The capacity of (i, j) is $Q_{ij}/N = \mu P_{ij}/N^2$ so the congestion is simply $\rho(F) = \mu/\min\{P_{ij}\}$. ∎

B. Almost Sure Convergence

The following proposition indicates that the shout gossip algorithms achieve consensus with probability one, *i.e.*, the shout gossip achieves consensus almost surely.

Proposition 1 (Almost sure convergence) *The probabilistic broadcast gossip algorithm converges, almost surely, to a consensus*

$$\mathsf{P}\left(\lim_{t \to \infty} x(t) = c\mathbf{1}\right) = 1 \tag{24}$$

for some random variable $c \in \mathbb{R}$.

Proof: We will make use of the following corollary to prove the almost sure convergence of the proposed algorithm [20] (the almost sure convergence of (possibly) non-stationary consensus algorithms with stochastic disturbances is given in [21], *i.e.*, almost sure convergence of linear consensus algorithms in its most general form).

Corollary 1 *Assume that for any $i \in \mathcal{V}$, we have that $W_{ii}(t) > 0$ almost surely. If $\mathcal{G}_W$ is strongly connected, then $W(t)$ achieves probabilistic consensus.*

We now from the definition of the algorithm that for all i, $W_{ii}(t) = 1 > 0$. Moreover, from Lemma 2, we have that

$$W_{jj} = 1 - \frac{1-\gamma}{N}\sum_{i=1, i \neq j}^{N} P_{ij} > \gamma \ , \tag{25}$$

implying that $W_{jj} > 0$, where the inequality in the above follows from the fact that $||P_{[j]}||_1 - P_{jj} < N$ since $P_{ij} < 1$. Moreover, we have, for $j \neq i$,

$$W_{ji} = \frac{1-\gamma}{N}P_{ji} > 0 \tag{26}$$

since $P_{ji} > 0$ and $\gamma < 1$. Thus, we have that $W_{ji} > 0$ for all i, j indicating that W is strongly connected and concluding the proof. ∎

C. Convergence in Expectation

We consider the convergence in expectation of the shout gossip algorithm. The next result reveals that, although the sum is not preserved per iteration, it is preserved in expectation.

We consider the initial state as deterministic, and hence all expectations are averaging the mixing matrices only.

The following theorem discusses the expectation of the limiting random vector, i.e., the expected value of $x(t)$ as t tends to infinity.

Proposition 2 (Convergence in expectation) *The expectation of the limiting random vector is given by*

$$\mathsf{E}\left\{\lim_{t\to\infty} x(t)\right\} = N^{-1}\mathbf{1}\mathbf{1}^T x(0). \qquad (27)$$

Proof: Suppose $|x_i(0)| \leq U < \infty$ from some U and all i. Recall that $W\mathbf{1} = \mathbf{1}$ and $\mathbf{1}^T W = \mathbf{1}^T$. Thus, W is non-expanding [22], [23] indicating that $|x_i(t)| \leq U$ for all $t \geq 0$ and i. Using dominated convergence theorem for bounded and converging random variables, we have

$$\mathsf{E}\left\{\lim_{t\to\infty} x(t)\right\} = \lim_{t\to\infty}\mathsf{E}\{x(t)\} = \lim_{t\to\infty} W^t x(0) \qquad (28)$$

since

$$x(t) = \prod_{k=0}^{t-1} W(k)x(0) \qquad (29)$$

$$\mathsf{E}\{x(t)\} = \prod_{k=0}^{t-1} \mathsf{E}\{W(k)\}x(0) \qquad (30)$$

$$\mathsf{E}\{x(t)\} = W^t x(0) \qquad (31)$$

where the first line follows from the fact that $W(t)$'s are independent and the second line is due to the fact that $\mathsf{E}\{W(t)\} = W$, $\forall t \geq 0$. We, then, need to characterize the limiting behavior of the average matrix W. The Perron-Frobenius theorem applied to stochastic matrices asserts that (provided all entries of W are strictly positive, see Lemma 2) the eigenvalue $\lambda_1(W) = 1$ is simple and all other eigenvalues of W satisfy $|\lambda_i(W)| < 1$, $\forall i = \{2, \ldots, N\}$. Thus [2],

$$\lim_{t\to\infty} W^t = N^{-1}\mathbf{1}\mathbf{1}^T \qquad (32)$$

concluding the proof of this Lemma. ∎

D. Convergence Time

We now turn to analyzing the time for the algorithm to reach consensus. To define this more precisely for our non-sum-preserving algorithms, we modify a definition of [2] which is also the definition given in [7].

Definition 1 (Convergence time) *For an $\epsilon > 0$, the ϵ-consensus time of an algorithm is the earliest time at which with probability $1 - \epsilon$ the deviation of the vector $x(k)$ from its mean, normalized by the initial deviation, is less than ϵ:*

$$T(N,\epsilon) = \sup_{x(0)}\inf\left\{t : \mathsf{P}\left(\frac{\|x(t) - Jx(t)\|_2}{\|x(0) - Jx(0)\|_2} \geq \epsilon\right) \leq \epsilon\right\}, \qquad (33)$$

where $\|\cdot\|_2$ denotes the ℓ_2 norm.

Proposition 3 (Consensus time) *The ϵ-consensus time of the algorithm is upper bounded:*

$$T(N,\epsilon) = O\left(\frac{\log \epsilon^{-1}}{\min\{P_{ij}\}}\right). \qquad (34)$$

Proof: We start with the methods of Boyd et al. [2, Lemma 2] applied to the deviation vector $\beta(t) = x(t) - Jx(t)$, as in the the non-sum preserving consensus algorithms of [7]:

$$\mathsf{E}[\|\beta(t)\|_2^2] \leq \lambda_1(W' - W'')^{2t}\|\beta(0)\|_2^2. \qquad (35)$$

Now Markov's inequality yields

$$\mathsf{P}\left(\frac{\|\beta(t)\|_2}{\|\beta(0)\|_2} \geq \epsilon\right) \leq \epsilon^{-2}\frac{\mathsf{E}[\|\beta(t)\|_2^2]}{\|\beta(0)\|_2^2} \qquad (36)$$

$$= \epsilon^{-2}\lambda_1(W' - W'')^{2t}. \qquad (37)$$

And therefore the ϵ-convergence time satisfies:

$$T(N,\epsilon) = O\left(\frac{\log \epsilon^{-1}}{\lambda_1(W' - W'')^{-1}}\right). \qquad (38)$$

Finally, we can apply Lemma 3:

$$T(N,\epsilon) = O\left(\frac{\log \epsilon^{-1}}{\min\{P_{ij}\}}\right), \qquad (39)$$

yielding the claimed result. ∎

IV. Discussion

For the simplified model of reception probability:

$$P_{ij} \propto \frac{\mathcal{P}}{D_{ij}^\beta}. \qquad (40)$$

In this model, for a network of N nodes with constant density,

$$\min\{P_{ij}\} \propto \frac{1}{N^{\beta/2}}, \qquad (41)$$

since the maximum distance for any pair of nodes scales like $\sqrt{N}$ and Lemma 3 gives the convergence bound

$$T(N,\epsilon) = O\left(N^{\beta/2}\log \epsilon^{-1}\right). \qquad (42)$$

If this reception probability is simply proportional to the expected received power so that the parameter β represent the path loss exponent, then we can see that for $\beta = 2$ the consensus time is linear in the number of nodes. This is perhaps unsurprising, since the expected number of connections at a distance $O(\sqrt{N})$ is constant, which means the graph is very well connected in expectation. For larger β the number of long-distance connections becomes smaller, leading to an increase in the consensus time.

We simulated the algorithm for a moderate-sized network of 100 nodes. Figure 1 shows the deviation from the consensus value as a function of the number of transmissions for different values of γ and β. The vertical axis is on a logarithmic scale. It is clear that as β increases the convergence is slower, but also that choosing larger γ leads to faster convergence.

However, choosing larger γ may lead to worse variance in the consensus value from the true mean, as shown in Figure 2.
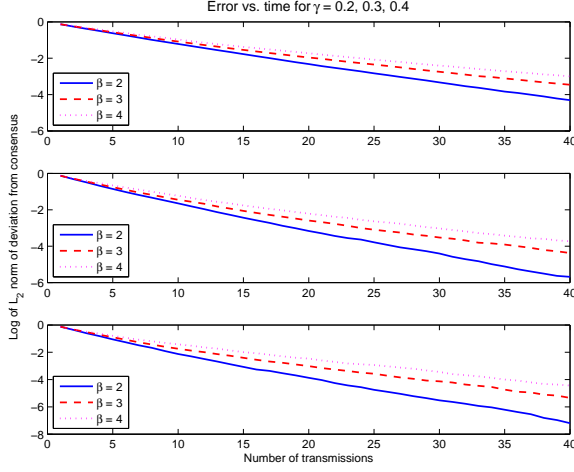
Fig. 1. Average squared distance from consensus value as a function of time for $\gamma = 0.2, 0.3, 0.4$. In each plot three curves are given for $\beta = 2, 3, 4$.

As γ increases, the variance of the consensus value increases significantly. This is because aggressive updating combined with the large neighborhoods given by smaller β result in more variance within the short time to convergence.
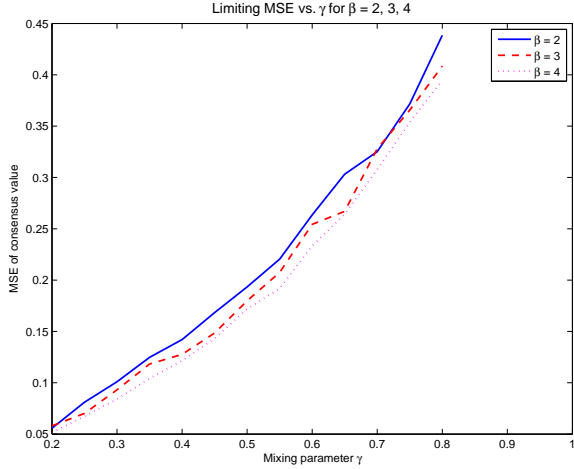

Fig. 2. MSE of consensus value from the true average as a function of γ.

It should be noted here that the bound from Lemma 3 can be greatly overestimating the convergence time for large path-loss exponents. This is because the routing in the Poincaré method is only using the direct links for any pair of nodes i, j which can have very small capacity for large β. It is easy to construct an alternative flow that routes on the nearest neighbor nodes on the grid with constant edge capacities, $N^{3/2}$ pairs of nodes using each edge and length $\sqrt{N}$ which yields the additional bound

$$T(N, \epsilon) = O\left(N^2 \log \epsilon^{-1}\right) \quad (43)$$

for any β, (and further for any Pij that is constant for constant distances) and therefore the previous convergence bound (42) should only be used for $\beta \leq 4$. An improved bounding

technique would involve a hybrid path routing between the two constructions we presented: using multi-hopping paths where the hop-length would be optimized as a function of the path-loss exponent β. This construction remains is left for the full version of this paper.

A key feature of our analysis is that the convergence time can be shown to depend on the probabilities Pij and that the bounds on the convergence time can be optimized for specific models of the connection probabilities. In intermediate-size networks broadcast-style algorithms may reach consensus faster than more complicated schemes based on pairwise averaging [7]. By adding in the possibility of a few long-range connections via the matrix P, the algorithm can converge much faster. A more complete analysis would include bounds on the error in consensus as in [7] and simulation results for different network topologies and signaling models.

APPENDIX

We gather the proofs of the technical lemmas here.

A. Proof of Lemma 1

Proof: We consider the proof of the first item. Let $\mathcal{E}_{[j]}(t)$ denote the event that $W_{[j]}\mathbf{1} = 1$. Note that

$$\mathsf{P}(W(t)\mathbf{1} = \mathbf{1}) = \mathsf{P}(\cap_{j=1}^{N}\mathcal{E}_{[j]}(t)). \quad (44)$$

Thus, we need to show that $\mathsf{P}(W_{[j]}\mathbf{1} = 1) = 1$, for all $j = 1, 2, \ldots, N$:

$$\mathsf{P}(\mathcal{E}_{[j]}(t)) = N^{-1}\sum_{i=1}^{N}\mathsf{P}(\mathcal{E}_{[j]}(t)|\mathcal{A}(t) = i) \quad (45)$$

$$= N^{-1}\sum_{i=1}^{N}\mathsf{P}(\mathcal{E}_{[j]}(t)|\mathcal{A}(t) = i, W_{[j]} = e_j)(1 - P_{ij})$$
$$+ \mathsf{P}(\mathcal{E}_{[j]}(t)|\mathcal{A}(t) = i, W_{[j]} = \gamma e_j + (1-\gamma)e_i)P_{ij} \quad (46)$$

$$= N^{-1}\sum_{i=1}^{N}P_{ij} + (1 - P_{ij}) \quad (47)$$

$$= 1 \quad (48)$$

since $\mathsf{P}(\mathcal{E}_{[j]}(t)|\mathcal{A}(t) = i, W_{[j]} = \gamma e_j + (1-\gamma)e_i) = 1$ and $\mathsf{P}(\mathcal{E}_{[j]}(t)|\mathcal{A}(t) = i, W_{[j]} = e_j) = 1$ as

$$(\gamma e_j + (1-\gamma)e_i)\mathbf{1} = e_j\mathbf{1} = \mathbf{1}. \quad (49)$$

This completes the proof of the first claim. Consider now $\mathcal{E}_{(j)}(t)$ denote the event that the j-th column of $W(t)$ sums to one, *i.e.*, $\mathbf{1}^T W_{(j)}(t) = 1$. Note that, for $j = i$,

$$\mathsf{P}(\mathcal{E}_{(j)}(t)) = N^{-1}\sum_{i=1}^{N}\mathsf{P}(\mathcal{E}_{(j)}(t)|\mathcal{A}(t) = i) \quad (50)$$

$$= N^{-1}\sum_{i=1}^{N}\prod_{j=1, j\neq i}^{N}(1 - P_{ij}) \quad (51)$$

since the event $\mathsf{P}(\mathcal{E}_{(j)}(t)|\mathcal{A}(t)=i)$ for $j=i$, equals unity if no node successfully receives the shouted value. Moreover,

$$\mathsf{P}(\mathcal{E}_{(j)}(t):j=i|\mathcal{A}(t)=i)=1 \tag{52}$$

$$\mathsf{P}(\mathcal{E}_{(j)}(t):j\neq i|\mathcal{A}(t)=i)=1 \tag{53}$$

$$\mathsf{P}(\cap_{j=1}^{N}\mathcal{E}_{(j)}(t))=1 \tag{54}$$

concluding the proof of the second claim. ∎

B. Proof of Lemma 2

Proof: Let $W(i,K)$ be the matrix $W(t)$ for transmitter i and collection of receivers K. We have

$$\mathsf{P}\left(W(t)=W(i,K)\right)=\prod_{j\in K}P_{ij}\prod_{j\in K^c\setminus\{i\}}(1-P_{ij}) . \tag{55}$$

From the definition of the state update,

$$W(i,K)=\gamma I+(1-\gamma)\operatorname{diag}(\mathbf{e}_{K^c})+(1-\gamma)\mathbf{e}_K\mathbf{e}_i^T \tag{56}$$

$$=I-(1-\gamma)\operatorname{diag}(\mathbf{e}_K)+(1-\gamma)\mathbf{e}_K\mathbf{e}_i^T . \tag{57}$$

This matrix has 1 on the diagonal elements corresponding to K^c and γ on the elements corresponding to K. The i-th column contains $(1-\gamma)$ in the rows corresponding to K. Note that $\mathsf{P}(j\in K\mid i)=P_{ij}$, which will make taking expectations easier. To calculate expectations we first take the expectation over K for fixed i and then the expectation over K.

Calculating W. Taking the expectation over K for fixed i, we get:

$$\mathsf{E}_K[W(i,K)\mid i]=I-(1-\gamma)\operatorname{diag}(\mathbf{p}_i)+(1-\gamma)\mathbf{p}_i\mathbf{e}_i^T \tag{58}$$

Then taking an expectation over i we get:

$$\mathsf{E}[W(t)]=I-\frac{1-\gamma}{N}\left(\operatorname{diag}\left(\sum_{i=1}^{N}\mathbf{p}_i\right)-P\right) \tag{59}$$

$$=I-\frac{1-\gamma}{N}L(P) . \tag{60}$$

Calculating W'. To calculate W' we first calculate $W(i,K)^TW(i,K)$:

$$W(i,K)^TW(i,K)$$
$$=I-2(1-\gamma)\operatorname{diag}(\mathbf{e}_K)+(1-\gamma)\left(\mathbf{e}_K\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_K^T\right)$$
$$+(1-\gamma)^2\operatorname{diag}(\mathbf{e}_K)$$
$$-(1-\gamma)^2\left(\mathbf{e}_K\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_K^T\right)+(1-\gamma)^2|K|\mathbf{e}_i\mathbf{e}_i^T$$
$$=I-(1-\gamma^2)\operatorname{diag}(\mathbf{e}_K)+\gamma(1-\gamma)\left(\mathbf{e}_K\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_K^T\right)$$
$$(1-\gamma)^2|K|\operatorname{diag}(\mathbf{e}_i) . \tag{61}$$

Now taking the expectation over K we obtain

$$\mathsf{E}_K[W(i,K)^TW(i,K)\mid i]$$
$$=I-(1-\gamma^2)\operatorname{diag}(\mathbf{p}_i)+\gamma(1-\gamma)\left(\mathbf{p}_i\mathbf{e}_i^T+\mathbf{e}_i\mathbf{p}_i^T\right)$$
$$+(1-\gamma)^2\|\mathbf{p}_i\|_1\operatorname{diag}(\mathbf{e}_i) . \tag{62}$$

Finally, taking the expectation over i:

$$\mathsf{E}[W(t)^TW(t)]$$
$$=I-\frac{1-\gamma^2}{N}\operatorname{diag}(P\mathbf{1})+\frac{\gamma(1-\gamma)}{N}(P+P^T)$$
$$+\frac{(1-\gamma)^2}{N}\operatorname{diag}(P\mathbf{1})$$
$$=I-\frac{2\gamma(1-\gamma)}{N}L(P) \tag{63}$$

Calculating W''. First we calculate $W(i,K)^T\mathbf{1}$:

$$W(i,K)^T\mathbf{1}=\mathbf{1}-(1-\gamma)\mathbf{e}_K+(1-\gamma)|K|\mathbf{e}_i . \tag{64}$$

This gives

$$W(i,K)^T\mathbf{1}\mathbf{1}^TW(i,K)$$
$$=\mathbf{1}\mathbf{1}^T-(1-\gamma)(\mathbf{1}\mathbf{e}_K^T+\mathbf{e}_K\mathbf{1}^T)$$
$$+(1-\gamma)|K|(\mathbf{1}\mathbf{e}_i^T+\mathbf{e}_i\mathbf{1}^T)$$
$$+(1-\gamma)^2\mathbf{e}_K\mathbf{e}_K^T$$
$$-(1-\gamma)^2|K|(\mathbf{e}_K\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_K^T)$$
$$+(1-\gamma)^2|K|^2\operatorname{diag}(\mathbf{e}_i) . \tag{65}$$

Let 1_{ij} denote the indicator variable node j receiving node i's transmission.. Now taking expectations over K:

$$\mathsf{E}[(\mathbf{1}\mathbf{e}_K^T+\mathbf{e}_K\mathbf{1}^T)\mid i]=(\mathbf{1}\mathbf{p}_i^T+\mathbf{p}_i\mathbf{1}^T) \tag{66}$$

$$\mathsf{E}[|K|(\mathbf{1}\mathbf{e}_i^T+\mathbf{e}_i\mathbf{1}^T)\mid i]=\|\mathbf{p}_i\|_1(\mathbf{1}\mathbf{e}_i^T+\mathbf{e}_i\mathbf{1}^T) \tag{67}$$

$$\mathsf{E}[\mathbf{e}_K\mathbf{e}_K^T\mid i]=\mathsf{E}\left[\sum_{j,k}1_{ij}1_{ik}\mathbf{e}_j\mathbf{e}_k^T\right]$$
$$=\mathbf{p}_i\mathbf{p}_i^T-\operatorname{diag}(\mathbf{p}_i\odot\mathbf{p}_i)$$
$$+\operatorname{diag}(\mathbf{p}_i) \tag{68}$$

$$\mathsf{E}[|K|(\mathbf{e}_K\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_K^T)\mid i]=\mathsf{E}\left[\sum_{j,k}1_{ij}1_{ik}(\mathbf{e}_k\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_k^T)\right]$$
$$=\sum_{j,k}P_{ij}P_{ik}(\mathbf{e}_k\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_k^T)-\sum_{j}P_{ij}^2(\mathbf{e}_j\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_j^T)$$
$$+\sum_{j}P_{ij}(\mathbf{e}_j\mathbf{e}_i^T+\mathbf{e}_i\mathbf{e}_j^T)$$
$$=\|\mathbf{p}_i\|_1\left(\mathbf{p}_i\mathbf{e}_i^T+\mathbf{e}_i\mathbf{p}_i^T\right)$$
$$-\left((\mathbf{p}_i\odot\mathbf{p}_i)\mathbf{e}_i^T+\mathbf{e}_i(\mathbf{p}_i\odot\mathbf{p}_i)^T\right)$$
$$+\left(\mathbf{p}_i\mathbf{e}_i^T+\mathbf{e}_i\mathbf{p}_i^T\right) \tag{69}$$

$$\mathsf{E}[|K|^2\operatorname{diag}(\mathbf{e}_i)\mid i]=\mathsf{E}\left[\sum_{j,k}1_{ij}1_{ik}\operatorname{diag}(\mathbf{e}_i)\right] \tag{70}$$
$$=\sum_{j,k}P_{ij}P_{ik}\operatorname{diag}(\mathbf{e}_i)-\sum_{j}P_{ij}^2\operatorname{diag}(\mathbf{e}_i)$$
$$+\sum_{j}P_{ij}\operatorname{diag}(\mathbf{e}_i)$$
$$=\left(\|\mathbf{p}_i\|_1^2-\|\mathbf{p}_i\odot\mathbf{p}_i\|_1+\|\mathbf{p}_i\|\right)\operatorname{diag}(\mathbf{e}_i) \tag{71}$$

Taking the expectation with respect to i yields:

$$N\mathsf{E}[(\mathbf{1}\mathbf{e}_K^T + \mathbf{e}_K\mathbf{1}^T)] = P^T\mathbf{1}\mathbf{1}^T P \tag{72}$$

$$N\mathsf{E}[|K|(\mathbf{1}\mathbf{e}_i^T + \mathbf{e}_i\mathbf{1}^T)] = P^T\mathbf{1}\mathbf{1}^T P \tag{73}$$

$$N\mathsf{E}[\mathbf{e}_K\mathbf{e}_K^T] = P^2 - \mathrm{diag}((P\odot P)\mathbf{1})$$
$$+ \mathrm{diag}(P\mathbf{1}) \tag{74}$$

$$N\mathsf{E}[|K|(\mathbf{e}_K\mathbf{e}_i^T + \mathbf{e}_i\mathbf{e}_K^T)] = \mathrm{diag}(P\mathbf{1})P + P\,\mathrm{diag}(P\mathbf{1})$$
$$+ 2P - 2(P\odot P) \tag{75}$$

$$N\mathsf{E}[|K|^2\,\mathrm{diag}(\mathbf{e}_i)] = \mathrm{diag}(P\mathbf{1})^2$$
$$- \mathrm{diag}((P\odot P)\mathbf{1})$$
$$+ \mathrm{diag}(P\mathbf{1})) \tag{76}$$

Putting it together:

$$\mathsf{E}[W(t)^T J W(t)]$$
$$= J + \frac{(1-\gamma)^2}{N^2}\Bigg(P^2 - \mathrm{diag}(P\mathbf{1})P - P\,\mathrm{diag}(\mathbf{1}P)$$
$$+ \mathrm{diag}(P\mathbf{1})^2 + 2\,\mathrm{diag}(P\mathbf{1})$$
$$+ 2P\odot P - 2P$$
$$- 2\,\mathrm{diag}((P\odot P)\mathbf{1})\Bigg) \tag{77}$$
$$= J + \frac{(1-\gamma)^2}{N^2}L(P)^2 + \frac{2(1-\gamma)^2}{N^2}L(P)$$
$$- \frac{2(1-\gamma)^2}{N^2}L(P\odot P) \tag{78}$$

concludes the proof. ∎

REFERENCES

[1] J. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Dept. of Electrical Engineering and Computer Science, M.I.T., Boston, MA, 1984.

[2] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508–2530, June 2006.

[3] A. G. Dimakis, A. D. Sarwate, and M. J. Wainwright, "Geographic gossip: Efficient aggregation for sensor networks," in *Proceedings of the Information Processing in Sensor Networks*, Nashville, TN, Apr. 2006.

[4] F. Benezit, A. Dimakis, P. Thiran, and M. Vetterli, "Gossip along the way : Order-optimal consensus through randomized path averaging," in *Proceedings of the 45th Annual Allerton Conference on Commununication, Control and Computation*, Sept. 2007.

[5] A. Nedic, A. Ozdaglar, and P. A. Parillo, "Constrained consensus and optimization in multi-agent networks," in *LIDS Tech report, 2779*, 2008.

[6] P. A. Bliman, A. Nedic, and A. Ozdaglar, "Rate of convergence for consensus with delays," in *IEEE CDC, 2008*, 2008.

[7] T. C. Aysal, M. E. Yildiz, A. D. Sarwate, and A. Scaglione, "Broadcast gossip algorithms for consensus," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, July 2009.

[8] B. Nazer, A. G. Dimakis, and M. Gastpar, "Local interference can accelerate gossip algorithms," in *Proceedings of the 46th Annual Allerton Conference on Commununication, Control and Computation*, Monticello, IL, 2008.

[9] ——, "Neighborhood gossip: Concurrent averaging through local interference," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2009)*, Taipei, Taiwan, 2009.

[10] D. Üstebay, B. Oreshkin, M. Coates, and M. Rabbat, "Greedy gossip with eavesdropping," Submitted to *IEEE Trans. Signal Processing*, March 2009.

[11] I. Schizas, A. Ribeiro, and G. Giannakis, "Consensus in ad hoc wsns with noisy links – part i: Distributed estimation of deterministic signals," *IEEE Trans. Signal Processing*, vol. 56, no. 1, pp. 350–364, Jan. 2008.

[12] F. F. P. Frasca, R. Carli and S. Zampieri., "Average consensus on networks with quantized communication," in *Submitted for publication*, 2008.

[13] S. Kar and J. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: Link failures and channel noise," *IEEE Trans. Signal Processing*, vol. 57, no. 1, pp. 355–369, Jan. 2009.

[14] A. G. Dimakis, A. D. Sarwate, and M. J. Wainwright, "Geographic gossip: Efficient averaging for sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 3, Mar. 2008.

[15] A. Molisch, *Wireless Communications*. John Wiley and Sons, 2005.

[16] P. Diaconis and D. Stroock, "Geometric bounds for eigenvalues of Markov chains," in *Annals of Applied Probability*, vol. 1, 1991.

[17] A. Sinclair, "Improved bounds for mixing rates of Markov chains and multicommodity flow," in *Combinatorics, Probability and Computing*, vol. 1, 1992.

[18] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.

[19] A. D. Sarwate and A. G. Dimakis, "The impact of mobility on gossip algorithms," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2009.

[20] F. Fagnani and S. Zampieri, "Randomized consensus algorithms over large scale networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 634 – 649, 2008.

[21] T. C. Aysal and K. E. Barner, "On the convergence of perturbed nonstationary consensus algorithms," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2009.

[22] T. C. Aysal, M. J. Coates, and M. G. Rabbat, "Distributed average consensus using dithered quantization," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4905–4918, Oct. 2008.

[23] ——, "Distributed average consensus using probabilistic quantization," in *Proc. IEEE Statistical Signal Processing Workshop*, Madison, WI, Aug. 2007.