Proceedings of the
46th IEEE Conference on Decision and Control
New Orleans, LA, USA, Dec. 12-14, 2007

Distributed Average Consensus with Stochastic Communication Failures

Stacy Patterson, Bassam Bamieh, and Amr El Abbadi

Abstract— **We consider a distributed average consensus algorithm over a network in which communication links fail with independent probability. Convergence in such stochastic networks is defined in terms of the variance of deviation from average. We characterize the decay factor of the variance in terms of the eigenvalues of a Lyapunov-like matrix recursion. We give expressions for the decay factors in the asymptotic limits of small failure probability and large networks. We also present a simulation-free method for computing the decay factor for any particular graph instance and use this method to study the behavior of various network examples as a function of link failure probability.**

I. INTRODUCTION

We consider the distributed average consensus problem over networks with stochastic communication failures. Each node has some initial value and the goal is for all nodes to reach consensus at the average of these initial values using only communication between neighbors in the network graph. Distributed average consensus is an important problem that has been studied in contexts such as vehicle formation [1], [2], [3], aggregation in sensor networks and peer-to-peer networks [4], and even load balancing in parallel processors [5], [6].

Distributed consensus has been widely investigated in static networks, where it has been shown that the convergence rate of a consensus algorithm depends on the second largest eigenvalue of the Laplacian of the network graph [7], [8]. However the assumption that a network topology is static, i.e. that communication links are fixed and reliable, is not always realistic. In sensor networks and mobile ad-hoc networks, for example, messages can be lost due to interference, and in wired networks, messages may be dropped due to buffer overflow. In these scenarios, it is desirable to quantify the effects these communication failures have upon the performance of the protocol.

Previous work on dynamic networks has focused on the identification of convergence conditions for distributed consensus algorithms. Conditions have been derived for dynamic or switching topologies where the network topology evolves over time [1], [2], [3]. It has also been shown that in a network with a time-varying topology, as long as the union of all infinitely occurring graph instances is connected, there is a distributed consensus algorithm that will eventually converge [9]. Work has also been done to establish convergence conditions for networks with stochastic communication failures. Hatano and Mesbahi [10] identify sufficient conditions for convergence in completely connected graphs, where each link has an equal probability of failure. The convergence condition depends on the second largest eigenvalue of the Laplacian of the mean network graph. The recent work by Kar and Moura [11] extends this model to include arbitrary topologies where links may fail with non-uniform probabilities and again establishes a sufficient condition based on the mean Laplacian.

This work is partly supported by NSF grant IIS 02-23022.

S. Patterson and A. El Abbadi are with the Department of Computer Science. B. Bamieh is with the Department of Mechanical Engineering. University of California, Santa Barbara, CA 93106-9560 USA. {sep,amr}@cs.ucsb.edu, bamieh@engineering.ucsb.edu

As already discussed, although there has been work that gives conditions for convergence with communication failures, to our knowledge, there has been no work to date that has quantified the effects of stochastic communication failures on the *convergence rate* of the distributed average consensus algorithm. We consider a network with an arbitrary, fixed underlying topology but where each edge fails with independent probability. In such stochastic networks, convergence is defined in terms of the variance of deviation from average. We characterize the decay factor of this variance in terms of the eigenvalues of a Lyapunov-like matrix recursion. We give expressions for the decay factors in the asymptotic limits of small failure probability and large networks. We also present a simulation-free method for computing the decay factor for any particular network instance and use this method to study the behavior of various network examples as a function of link failure probability.

The remainder of this paper is organized as follows. In Section II, we define our system model and distributed consensus algorithm. Section III gives our main convergence results. Section IV presents computational results on decay factors for various network topologies. Finally, we conclude in Section V.

II. PROBLEM FORMULATION

We model the network as an undirected graph $G = (V, E)$ where V is the set of nodes, with $|V| = n$, and E is the set of communication links between them. In this work, we assume that each link $(i, j) \in E$ has an independent probability $p_{(i,j)}$ of failing in each round. If a link fails, no communication takes place across the link in either direction in that round. A link that does not fail in round k is *active*. The neighbor set of node i, denoted by $N_i(k)$ for round k, is the set of nodes with which node i has active communication links in round k.

We consider the following simple distributed consensus algorithm. Every node i has an initial value $x_i(0)$, and the average of all values in the system is $x_{ave} = \frac{1}{n}\sum_{i=1}^{n} x_i(0)$. The objective of the algorithm is to converge to an equilibrium where $x_i(k) = x_{ave}$ for all $i \in V$. In each round, each node sends a fraction β of its current value to each neighbor with which it has an active communication link. Each node's value is updated according to the following rule.

$$x_i(k+1) = \beta \sum_{j \in N_i(k)} x_j(k) + (1 - \beta|N_i(k)|)x_i(k),$$

where β is the parameter that defines an instance of the algorithm. This algorithm can be implemented without any *a priori* knowledge of link failures.

In a network with no communication failures, this algorithm can be expressed as an n by n matrix, $A := I - \beta\mathcal{L}$, where $\mathcal{L}$ is the Laplacian matrix[1] of the graph G. The evolution of the system is described by the following recursion equation.

$$x(k+1) = Ax(k) \qquad (1)$$

It is a well known result that for $0 < \beta \leq \frac{1}{d_{max}}$ (with equality only if the graph is not bipartite), where d_{max} is the maximum degree of G, the system converges to equilibrium at x_{ave} if and only if the magnitude of the second largest eigenvalue of A, $\lambda_2(A)$, is strictly less than 1 [8]. The convergence rate of the system can be computed from $\lambda_2(A)$.

(1) can be extended to include stochastic communication failures as follows. Let $b_{(i,j)}$ be the vector with the i'th entry equal to 1, the j'th entry equal to -1 and all other entries equal to 0. $B_{(i,j)}$ is defined as

$$B_{(i,j)} := \beta\, b_{(i,j)} b^*_{(i,j)}. \qquad (2)$$

The system can then be described by the following recursion equation.

$$x(k+1) = \left(A + \sum_{(i,j) \in E} \delta_{(i,j)}(k) B_{(i,j)} \right) x(k) \qquad (3)$$

where $\delta_{(i,j)}$ is a Bernoulli random variable with

$$\delta_{(i,j)}(k) \;:=\; \begin{cases} 1 & \text{with probability } p_{(i,j)} \\ 0 & \text{with probability } 1 - p_{(i,j)} \end{cases}$$

When $\delta_{(i,j)} = 1$, the edge (i,j) has failed. One can interpret (3) as first performing the algorithm on the complete communication graph, and then simulating the failed edges by undoing the effects of communication over those edges. In essence, each $B_{(i,j)}$ matrix returns the values sent across edge (i,j), yielding the state in which edge (i,j) did not play a part.

We rewrite (3) in a form that is more convenient for our analysis using zero mean random variables. Let $\mu_{(i,j)}(k) := \delta_{(i,j)}(k) - p_{(i,j)}$ and observe that they are zero mean.

The dynamics can now be rewritten as

$$x(k+1) = \overline{A}x(k) + \sum_{(i,j) \in E(k)} \mu_{(i,j)}(k) B_{(i,j)} x(k) \qquad (4)$$

where $\bar{A} := A + \sum_{(i,j) \in E} p_{(i,j)} B_{(i,j)}$.

[1] Let $\mathcal{E}$ be the adjacency matrix of a graph G and $\mathcal{D}$ be the diagonal matrix with the diagonal entry in row i equal to the degree of node i. Then the Laplacian matrix of G is defined as $\mathcal{L} := \mathcal{D} - \mathcal{E}$.

We measure how far the current state of the system is from the average of all states using the *deviation from average* vector $\tilde{x}$ whose components are

$$\tilde{x}_i(k) \;:=\; x_i(k) \;-\; \frac{1}{n}\left(x_1(k) \;+\; \ldots \;+\; x_n(k)\right).$$

The entire vector $\tilde{x}$ can be written as the projection

$$\tilde{x}(k) \;=\; \mathcal{P}\; x(k),$$

with $P := (I - \frac{1}{n}\mathbf{1}\mathbf{1}^*)$, where $\mathbf{1}$ is the vector with all entries of 1.

In this paper, we are primarily interested in characterizing the convergence rate of $\tilde{x}$ to zero. Since the dynamics of x and $\tilde{x}$ are stochastic, we use the decay rate of the *variance of total deviation from average* $\mathcal{E}\left\{||\tilde{x}(k)||^2\right\}$ as an indicator of the rate of convergence.

Problem Statement: *Consider a distributed consensus algorithm with random link failures as modeled by the system with multiplicative noise (4). Determine the rate at which the variance of total deviation from average* $\mathcal{E}\left\{||\tilde{x}(k)||^2\right\}$ *converges to* 0 *as* $k \to \infty$.

The key to addressing this problem is to study the equations governing the second order statistics of the states of (4). To this end, we define the autocorrelation matrices of x and $\tilde{x}$ by

$$\begin{aligned} M(k) &:= \mathcal{E}\left\{x(k)x^*(k)\right\}, \\ \tilde{M}(k) &:= \mathcal{E}\left\{\tilde{x}(k)\tilde{x}^*(k)\right\}, \end{aligned}$$

and note that they are related by the projection $\mathcal{P}$

$$\begin{aligned} \tilde{M}(k) &= \mathcal{E}\left\{\tilde{x}(k)\tilde{x}^*(k)\right\} \;=\; \mathcal{E}\left\{\mathcal{P}x(k)x^*(k)\mathcal{P}\right\} \\ &= \mathcal{P}\;\mathcal{E}\left\{x(k)x^*(k)\right\}\;\mathcal{P} \\ &= \mathcal{P}\;M(k)\;\mathcal{P}. \end{aligned}$$

The variance of the total deviation from average is given by the trace of $\tilde{M}$, since

$$\begin{aligned} \mathcal{E}\left\{||\tilde{x}(k)||^2\right\} &= \mathcal{E}\left\{\tilde{x}^*(k)\;\tilde{x}(k)\right\} \;=\; \mathrm{tr}\left(\mathcal{E}\left\{\tilde{x}(k)\tilde{x}^*(k)\right\}\right) \\ &= \mathrm{tr}\left(\tilde{M}(k)\right). \end{aligned}$$

It is well known that the autocorrelation matrix of the system (4) with zero-mean multiplicative noise [12] is given by the following recursion equation

$$M(k+1) = \overline{A}M(k)\overline{A} + \sum_{(i,j) \in E} \sigma^2_{(i,j)} B_{(i,j)} M(k) B_{(i,j)} \qquad (5)$$

where $\sigma^2_{(i,j)} := \mathrm{var}(\mu_{(i,j)}(k))$. This is a discrete-time Lyapunov-like matrix difference equation. The additional terms multiplying $\sigma^2_{(i,j)}$ in (5) makes this a nonstandard Lyapunov recursion. The matrix $\tilde{M}(k)$ satisfies a similar recursion relation which we derive in the next section and then study its convergence properties.

III. Characterizing Convergence

In this section, we first derive a recursion equation for $\tilde{M}$, the autocorrelation of $\mathcal{E}\left\{||\tilde{x}(k)||^2\right\}$ which is the variance of the total deviation from average. We then characterize the decay rate of this variance in terms of the eigenvalues of a Lyapunov-like matrix valued operator. An exact computational procedure for these eigenvalues is given in the next section, while we give expressions in this section for the asymptotic cases of small failure probability p and large network size n.

For simplicity, we assume that all edges have equal failure of probability, p. Therefore $\sigma^2_{(i,j)} = \sigma^2 = p - p^2$ for all $(i,j) \in E$. The convergence results can easily be generalized to the non-uniform probability failure model.

Lemma 3.1: The matrices $\tilde{M}(k)$ (under the uniform probability assumption) satisfy the recursion

$$\begin{aligned}\tilde{M}(k+1) &= \left(\tilde{A} + p\beta\mathcal{L}\right)\tilde{M}(k)\left(\tilde{A} + p\beta\mathcal{L}\right) \\ &\quad + \sigma^2 \sum_{(i,j)\in E} B_{(i,j)}\tilde{M}(k)B_{(i,j)}, \qquad (6)\end{aligned}$$

where $\tilde{A} := \mathcal{P}A\mathcal{P}$.

Proof:

First note that from the definitions of the matrices $B_{(i,j)}$, their sum is proportional to the graph's Laplacian, *i.e.* $\sum_{(i,j)\in E} B_{(i,j)} = \beta\mathcal{L}$. Therefore, $\bar{A}$ is simply

$$\begin{aligned}\bar{A} &= A + p\sum_{(i,j)\in E} B_{(i,j)} = A + p\beta\,\mathcal{L} \\ &= I - (1-p)\beta\,\mathcal{L}.\end{aligned}$$

Observe that the following equalities hold for the action of $\mathcal{P}$ on any of the matrices $B_{(i,j)}$

$$B_{(i,j)}\mathcal{P} = \beta\, b_{(i,j)}b^*_{(i,j)}(I - \frac{1}{n}\mathbf{1}\mathbf{1}^*) = \beta\, b_{(i,j)}b^*_{(i,j)} = B_{(i,j)},$$

where the second equality follows from $\mathbf{1}^* b_{(i,j)} = 0$ for any edge (i,j). Similarly $\mathcal{P}\,B_{(i,j)} = B_{(i,j)}$.

The second fact needed is that $\mathcal{L}$, and consequently A and $\bar{A}$, commute with the projection $\mathcal{P}$. This follows from the fact that $\mathbf{1}$ is both a left and a right eigenvector of $\mathcal{L}$.

(6) follows from multiplying both sides of (5) by $\mathcal{P}$ and using $\mathcal{P} = \mathcal{P}^2$ as follows

$$\begin{aligned}&\tilde{M}(k+1) \\ &= \mathcal{P}M(k+1)\mathcal{P} \\ &= \mathcal{P}\overline{A}M(k)\overline{A}\mathcal{P} + \sigma^2\sum_{(i,j)\in E}\mathcal{P}B_{(i,j)}M(k)B_{(i,j)}\mathcal{P} \\ &= \mathcal{P}^2\overline{A}M(k)\overline{A}\mathcal{P}^2 + \sigma^2\sum_{(i,j)\in E}B_{(i,j)}\mathcal{P}M(k)\mathcal{P}B_{(i,j)} \\ &= \mathcal{P}\overline{A}\mathcal{P}M(k)\mathcal{P}\overline{A}\mathcal{P} + \sigma^2\sum_{(i,j)\in E}B_{(i,j)}\tilde{M}(k)B_{(i,j)} \\ &= \mathcal{P}\overline{A}\mathcal{P}^2M(k)\mathcal{P}^2\overline{A}\mathcal{P} + \sigma^2\sum_{(i,j)\in E}B_{(i,j)}\tilde{M}(k)B_{(i,j)}\end{aligned}$$

$$\begin{aligned}\tilde{M}(k+1) &= \left(\tilde{A} + p\beta\mathcal{L}\right)\tilde{M}(k)\left(\tilde{A} + p\beta\mathcal{L}\right) \\ &\quad + \sigma^2\sum_{(i,j)\in E}B_{(i,j)}\tilde{M}(k)B_{(i,j)}.\end{aligned}$$

■

To study the decay or growth properties of the matrix sequence $\tilde{M}(k)$, we define the Lypunov-like operator

$$\begin{aligned}\mathcal{A}(X) &:= \left(\tilde{A} + p\beta\mathcal{L}\right)X\left(\tilde{A} + p\beta\mathcal{L}\right) \\ &\quad + (p - p^2)\sum_{(i,j)\in E}B_{(i,j)}\;X\;B_{(i,j)}. \qquad (7)\end{aligned}$$

The linear matrix recursion (6) can now be written as

$$\tilde{M}(k+1) = \mathcal{A}\left(\tilde{M}(k)\right). \qquad (8)$$

Since this is a linear matrix equation, the condition for asymptotic decay of each entry of $\tilde{M}(k)$ is

$$|\rho(\mathcal{A})| < 1,$$

where $\rho(\mathcal{A})$ is the spectral radius of $\mathcal{A}$, which we call the *decay factor* of the algorithm instance. In fact, since each entry of $\tilde{M}(k)$ has the asymptotic bound of a constant times $|\rho(\mathcal{A})|^k$, then so does its trace and consequently $\mathcal{E}\left\{||\tilde{x}(k)||^2\right\}$.

We summarize these results in the following theorem.

Theorem 3.2: Consider a distributed consensus algorithm with random link failures as modeled by the system with multiplicative noise (4).

1) The variance of the total deviation from average $\mathcal{E}\left\{||\tilde{x}(k)||^2\right\}$ converges to 0 as $k \to \infty$ if and only if
$$|\rho(\mathcal{A})| < 1.$$
2) The worst case asymptotic growth of $\mathcal{E}\left\{||\tilde{x}(k)||^2\right\}$ is given by
$$|\rho(\mathcal{A})|^k.$$

Note that in the case that links do not fail, when $p = 0$, we have

$$\mathcal{A}:\quad X \;\mapsto\; \tilde{A}X\tilde{A}$$

and $\rho(\mathcal{A})$ is precisely $(\rho(\tilde{A}))^2$, which is the square of the eigenvalue of A with the second largest modulus, as is well known. However, when failures occur with non-zero probability, $p > 0$, the additional terms in the operator $\mathcal{A}$ play a role. For $p \neq 0$, the operator $\mathcal{A}$ is no longer a pure Lyapunov operator of the form $X \mapsto \tilde{A}X\tilde{A}$ but rather a sum of such terms. Thus, one does not expect a simple relationship between the eigenvalues of $\mathcal{A}$ and those of the constitutive matrices as in the pure Lyapunov operator case.

Perturbation Analysis

One important asymptotic case is that of small, uniform link failure probability p. We can analyze this case by doing a first order eigenvalue perturbation analysis of the operator $\mathcal{A}$ in (7) as a function of the parameter p. We first recall the basic set up from analytic perturbation theory for eigenvalues of symmetric operators [13].

Consider a symmetric, matrix-valued function $\mathcal{A}(p, X)$ of a real parameter p and matrix X of the form

$$\mathcal{A}(p, X) \;=\; \mathcal{A}_o(X) \;+\; p\,\mathcal{A}_1(X) \;+\; p^2\,\mathcal{A}_2(X).$$

Let $\gamma(p)$ and $W(p)$ be an eigenvalue-eigenmatrix pair of $\mathcal{A}(p, .)$ as p varies, i.e.

$$\mathcal{A}(p, W(p)) \;=\; \gamma(p) W(p).$$

It is a standard result of spectral perturbation theory that for isolated eigenvalues of $\mathcal{A}(0, .)$ the functions γ and W are well defined and analytic in some neighborhood $p \in (-\epsilon, \epsilon)$.

The power series expansion of γ is

$$\gamma(p) \;=\; \lambda \;+\; c_1 p \;+\; c_2 p^2 \;+\; \cdots,$$

where λ is an eigenvalue of $\mathcal{A}_o$. The calculation of the coefficient c_1 involves the eigenmatrix V of λ and is given by

$$c_1 \;=\; \frac{\langle V, \mathcal{A}_1(V)\rangle}{\langle V, V\rangle}. \tag{9}$$

Note that we are dealing with matrix-valued operators on matrices, and the inner product on matrices is given by $\langle X, Y\rangle \;:=\; \mathrm{tr}\,(X^* Y)$.

In order to apply this procedure to the operator $\mathcal{A}$ in (7), we first note that it can be written as

$$\mathcal{A} \;=\; \mathcal{A}_o \;+\; p\,\mathcal{A}_1 \;+\; p^2\,\mathcal{A}_2,$$

where

$$\begin{aligned}
\mathcal{A}_o(X) &= \tilde{A} X \tilde{A} \\
\mathcal{A}_1(X) &= \beta \mathcal{L} X \tilde{A} \;+\; \beta \tilde{A} X \mathcal{L} \;+\; \sum_{(i,j)\in E} B_{(i,j)}\; X\; B_{(i,j)} \\
\mathcal{A}_2(X) &= \beta^2 \mathcal{L} X \mathcal{L} \;-\; \sum_{(i,j)\in E} B_{(i,j)}\; X\; B_{(i,j)}.
\end{aligned}$$

Now to investigate the first order behavior of the largest eigenvalue, we observe that the eigenmatrix corresponding to the largest eigenvalue of $\mathcal{A}_o$ is

$$V \;=\; v_2 v_2^*,$$

where v_2 is the vector corresponding to the second smallest eigenvalue of the Laplacian $\mathcal{L}$, also called the Fiedler vector. Applying formula (9) to this expression for V yields the first order term in the expansion of the largest eigenvalue of $\mathcal{A}$ to be

$$c_1 \;=\; 2\beta\; \underline{\lambda}(\mathcal{L})\; \overline{\lambda}(\tilde{A}) \;+\; \sum_{(i,j)\in E} \left(v_2^* B_{(i,j)} v_2\right)^2, \tag{10}$$

where $\underline{\lambda}(\mathcal{L})$ is the second smallest eigenvalue of $\mathcal{L}$ and $\overline{\lambda}(\tilde{A})$ is the largest eigenvalue of $\tilde{A}$ (equivalently, the second largest eigenvalue of A).

Since $A = I - \beta\mathcal{L}$ it follows that [14], [15] for $\beta \le \frac{1}{d_{\max}}$ we have

$$\overline{\lambda}(\tilde{A}) \;=\; 1 - \beta\underline{\lambda}(\mathcal{L}).$$

Using this identity, the fact that $\rho(\mathcal{A}_o) \;=\; (\rho(\tilde{A}))^2 \;=\; (1 - \beta\underline{\lambda}(\mathcal{L}))^2$, and equation (10) above gives the following expression for $\rho(\mathcal{A})$ which is valid up to first order in p

$$\begin{aligned}
\rho(\mathcal{A}) \;&=\; 1 \;-\; 2(p-1)\beta\;\underline{\lambda}(\mathcal{L}) \;+\\
&\quad (1-2p)\beta^2(\underline{\lambda}(\mathcal{L}))^2 \;+\; p\sum_{(i,j)\in E}\left(v_2^* B_{(i,j)} v_2\right)^2.
\end{aligned}$$

In the special case of a tori network, there are explicit asymptotic expressions [14], [15] for $\underline{\lambda}(\mathcal{L})$. Furthermore, it can be shown that the term quadratic in v_2 decays to zero with the size of the network. These facts lead to the following statement whose proof we omit for brevity.

Theorem 3.3: For d-dimensional tori of size n, the first order expansion (in p) of the decay factor is given by

$$\rho(\mathcal{A}) \;=\; 1 \;-\; (p-1)\beta\frac{8\pi^2}{n^{2/d}} \;+\; (1-2p)O\left(\frac{1}{n^{4/d}}\right).$$

It is interesting to note that for large n, the leading order behavior of the decay factor is

$$1 \;-\; (p-1)\beta\frac{8\pi^2}{n^{2/d}}.$$

Recall that β is the fraction that is sent across each link. Therefore for large n, link failures will reduce the effective fraction that is sent across each link by a factor of $1-p$.

IV. Computations

In this section, we give computational results for the multiplicative decay factor, $\rho(\mathcal{A})$, of various network topologies as function of uniform link failure probability.

Since we have a characterization of the decay factor in terms of the largest eigenvalue of the linear operator $\mathcal{A}$ defined in (7), it is not necessary to perform Monte Carlo simulations of the original system (4) to compute decay factors. However, $\mathcal{A}$ is not in a form to which standard eigenvalue computation routines (such as those in MATLAB) can be immediately applied. We present a simple procedure to obtain a matrix representation of $\mathcal{A}$ which can then be readily used in eigenvalue computation routines.

Recall that the Kronecker product of any two $m\times n$ and $r\times s$ matrices C and D respectively is the $mr\times ns$ matrix

$$C \otimes D \;:=\; \begin{bmatrix} c_{11}D & \cdots & c_{1n}D \\ \vdots & \ddots & \vdots \\ c_{m1}D & \cdots & c_{mn}D \end{bmatrix}.$$

Let $vec(X)$ denote the "vectorization" of any $m\times n$ matrix X constructed by stacking the matrix columns on top of one another to form an $mn\times 1$ vector. It then follows that a matrix equation of the form $Y = CXD$ can be rewritten using matrix-vector products as

$$vec(Y) \;=\; (C\otimes D)\; vec(X).$$

Thus, using Kronecker products, $\mathcal{A}$ in (7) has a matrix representation of the form

$$\mathcal{A} \;=\; \left(\tilde{A} + p\beta\mathcal{L}\right)\otimes\left(\tilde{A} + p\beta\mathcal{L}\right) + \sigma^2\sum_{(i,j)\in E} B_{(i,j)}\otimes B_{(i,j)}.$$

For a graph with n nodes, $\mathcal{A}$ is an $n^2 \times n^2$ matrix. This matrix representation can be used to find $\rho(\mathcal{A})$ via readily available eigenvalue routines in MATLAB.

We do this next for several examples and investigate the behavior of the decay factor $\rho(\mathcal{A})$ as a function of the probability of link failure. For each topology, we compute the decay factor for several values of β, including the value which is optimal for each graph when there are no communication failures. This value is given by the following [8],

$$\beta^* = \frac{2}{\underline{\lambda}(\mathcal{L}) + \overline{\lambda}(\mathcal{L})},$$

where $\underline{\lambda}(\mathcal{L})$ and $\overline{\lambda}(\mathcal{L})$ are the second smallest and the largest eigenvalues of the Laplacian matrix of the graph, respectively.

A. Decay Factors for Tori

Figures 1 and 2 give the decay factors for a ring network with 9 nodes and a 2-dimensional discrete torus with 25 nodes. For each topology, we compute the decay factors using the maximum β that guarantees convergence, $\beta := \frac{1}{d_{max}}$, the optimal β, and a smaller $\beta := \frac{1}{2d_{max}}$. For the ring network, the maximum β is 0.5, the optimal β is approximately 0.4601, and the smaller β is 0.25. For the 2-dimensional torus, the maximum β is 0.25, the optimal β is approximately 0.2321 and the smaller β is 0.125. We elect to use small networks because the difference between the optimal β and maximal β is more noticeable.

As expected, in both networks, when there are no link failures, the decay factor is smallest for the optimal β. Surprisingly, for the maximum β, the decay factors decrease for small probabilities of failure. We conjecture that the failures reduce the effective fractions of values that are sent across each edge over a large number of rounds. As the probability of failure increases, the effective fraction decreases to approach the optimal β, and thus the algorithm performance actually improves. The decay factor continues to decrease until the failure probability reaches approximately 0.1 and then steadily increases. For the case where $\beta := \frac{1}{2d_{max}}$, the edge weight is less than the optimal, and so introducing failures only increases the decay factor. These results demonstrate that there is a relationship between the failure probability and the choice of β, and therefore it seems possible to select a β that optimizes performance for a given failure probability.

B. Decay Factors for Random Graphs

We also compute the decay factors for two random graph topologies. Fig. 3 shows results for an Erdős-Rényi (ER) random graph [16] of 50 nodes, where each pair of nodes is connected with probability 0.25. The graph has 287 edges and a maximum node degree of 20. The maximum β is $\frac{1}{2d_{max}} = \frac{1}{20}$. The optimal β is approximately 0.0724. We also show a β that is less than optimal, $\beta := \frac{1}{2d_{max}} = \frac{1}{40}$. Fig. 4 shows results for a scale free network of 50 nodes generated using the BA Model [17]. The distribution of node degrees obeys a power law, such as has been observed in the

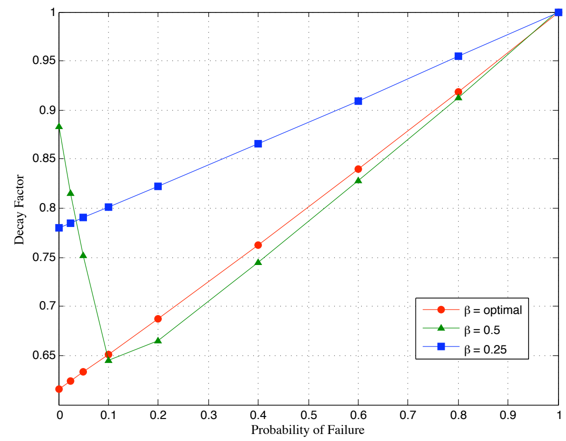

Fig. 1. Decay Factor for Various Link Failure Probabilities in a 9 Node Ring Network

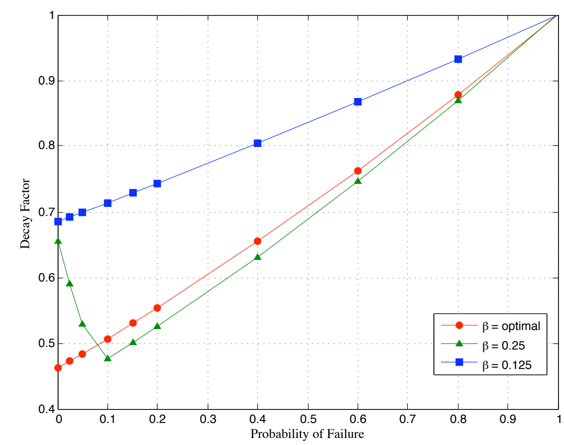

Fig. 2. Decay Factor for Various Link Failure Probabilities in a 25 Node 2-Dimensional Torus

Internet, with a maximum node degree of 25. The graph has 394 edges. The maximum β is $\frac{1}{25}$. The optimal β is approximately 0.0787. We also show a β that is less than optimal, $\beta := \frac{1}{50}$.

Again, in both networks, the optimal β yields the smallest decay factor when there is zero probability of edge failure. However, unlike in the ring and 2-dimensional networks, the maximum β does not result in a performance improvement for small failure probabilities.

It has been shown that there is an assignment of fractions, or weights, to graph edges that will yield the smallest decay factor [8]. In tori networks, this assignment is equivalent to assigning the optimal β to every edge. In arbitrary networks, the optimal edge weight assignment may not have equal weights on every edge. We believe that in a random network with the maximum β value, introducing edge failures with uniform probability does not give any performance improvement because the effective fraction associated with each edge is decreased uniformly. Thus, the addition of failures does not change the effective edge weights in a manner that

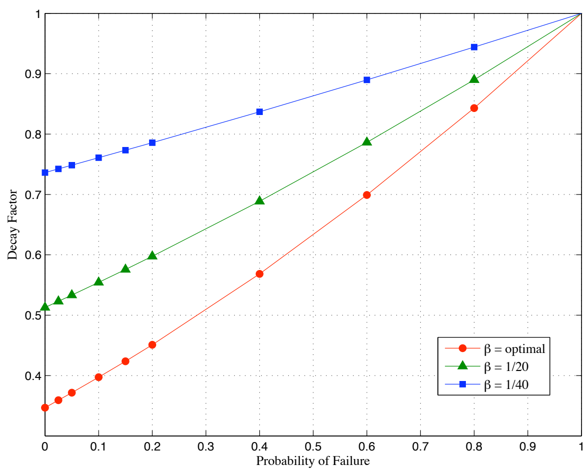

Fig. 3. Decay Factor for Various Link Failure Probabilities in a 50 Node ER Graph

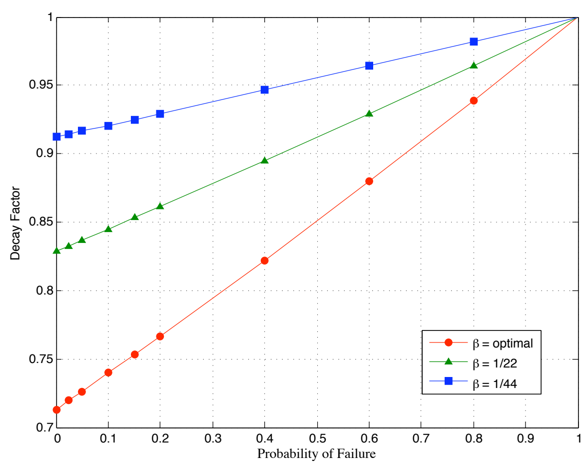

Fig. 4. Decay Factor for Various Link Failure Probabilities in a 50 Node Scale Free Graph

approaches the optimal weight assignment.

V. Conclusions

We have presented an analysis of a distributed average consensus algorithm in networks with stochastic communication failures. We have shown that the convergence rate of the consensus algorithm can be characterized by the largest eigenvalue of a Lyapunov-like matrix recursion, and we have developed expressions for the multiplicative decay factor in the asymptotic limits of small failure probability and large networks. We have also shown that the decay factor can be computed using a simulation-free method. Using this method, we have computed the decay factors for various network topologies for increasing failure probabilities. These computations indicate that there is a relationship between the network topology, the algorithm parameter β, and the probability of failure that is more complex than intuition would suggest. In particular, we show that for certain network topologies, communication failures can actually improve algorithm performance.

As the subject of current work, we are investigating the extension of our model and analysis to incorporate communication failures that are spatially and temporally correlated. Such extensions will allow us to study more realistic network conditions such as network partitions and node failures.

References

[1] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," in *IEEE Transactions on Automatic Control*, vol. 48, June 2003, pp. 988–1001.
[2] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," in *IEEE Transactions on Automatic Control*, vol. 49, September 2004, pp. 1520–1533.
[3] L. Moreau, "Stability of multiagent systems with time-dependent communication links," in *IEEE Transactions on Automatic Control*, vol. 50, February 2005, pp. 169–182.
[4] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," in *Systems and Control Letters*, vol. 53, 2005, pp. 65–78.
[5] G. Cybenko, "Dynamic load balancing for distributed memory multiprocessors," *Journal of Parallel and Distributed Computing*, vol. 7, no. 2, pp. 279–301, 1989.
[6] J. E. Boillat, "Load balancing and poisson equation in a graph," *Concurrency: Practice and Experience*, vol. 2, no. 4, pp. 289–313, 1990.
[7] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.
[8] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 52, pp. 65–78, 2004.
[9] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Information Processing in Sensor Networks (IPSN 2006)*, 2005, pp. 63–70.
[10] Y. Hatano and M. Mesbahi, "Agreement over random networks," *IEEE Transactions on Automatic Control*, vol. 50, pp. 1867–1872, November 2005.
[11] S. Ka and J. Moura, "Distributed average consensus in sensor networks with random link failures," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2007.
[12] S. Boyd, L. El Ghauoi, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, ser. Studies in Applied Mathematics. SIAM, 1994, vol. 15.
[13] H. Baumgarte, *Analytic perturbation theory for matrices and operators*. Birkhauser Verlag, Basel, 1985.
[14] S. Patterson, B. Bamieh, and A. El Abbadi, "Brief announcement: Convergence analysis of scalable gossip protocols," in *20th International Symposium on Distributed Computing*, 2006, pp. 540–542.
[15] ——, "Convergence analysis of scalable gossip protocols," in *Technical Report 2006-09*, July 2006.
[16] P. Erdos and A. Renyi, "On the evolution of random graphs," *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, vol. 5, pp. 17–61, 1960.
[17] R. Albert and A.-L. Barabasi, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, pp. 47–97, 2002.