Distributed Probabilistic Synchronization Algorithms for Communication Networks

Mehmet Akar and Robert Shorten

Abstract—In this paper, we present a probabilistic synchronization algorithm whose convergence properties are examined using tools of row-stochastic matrices. The proposed algorithm is particularly well suited for wireless sensor network applications, where connectivity is not guaranteed at all times, and energy efficiency is an important design consideration. The tradeoff between the convergence speed and the energy use is studied.

Index Terms—Common Lyapunov function, consensus, scrambling matrix, switched systems, synchronization.

I. INTRODUCTION

Synchronization phenomena have been a topic of research in the physical sciences and in mathematics for quite some time [1]–[26]. This interest has been primarily motivated by applications that require networks of agents to acquire a common state. Such applications are pervasive, and include clock synchronization in computer networks [1]–[7], coordination of unmanned air vehicles [20]–[24], and allocation of network resources fairly [27].

II. PROBLEM DESCRIPTION AND MATHEMATICAL PRELIMINARIES

Let \( x(t) = [x_1(t), x_2(t), \ldots, x_n(t)]^T \in \mathbb{R}^n \) denote the values of some network-wide variable for \( n \) agents. The objective is to achieve a common value for all nodes, ideally \( x_1(t) = x_2(t) = \ldots = x_n(t) \) for all \( t \), which will be referred to as nodes being synchronized in this paper. Clearly, this requires at least a single node transmitting to all others for all time; an assumption that usually fails in mobile ad hoc and sensor networks due to the lack of connectivity at certain time intervals. Therefore, we are interested in distributed algorithms that achieve synchronization at least asymptotically [i.e., \( x_i(t) \to c \) as \( t \to \infty \)] that can be tackled by using the well-known averaging ideas first studied by Markov [16]. To this end, let each agent update its value according to

\[
x_i(t+1) = \sum_{j=1}^{n} w_{ij}(t) x_j(t)
\]

where \( w_{ij}(t) \) are nonnegative averaging coefficients that satisfy

\[
w_{ij}(t) \geq 0 \quad \forall i, j, t, \quad \sum_{j=1}^{n} w_{ij}(t) = 1 \quad \forall i, t.
\]

As we shall see in Section III, the weights used by each node depend only on the number of receptions.

A. Graph Representation

At each time \( t \), we can associate a graph \( [V, E(t)] \) with (1), where \( V = \{1, 2, \ldots, n\} \) is the set of vertices, \( E(t) \) is the set of directed edges, and \( (j, i) \in E(t) \) holds if and only if \( w_{ij}(t) > 0 \) (i.e., there is communication from node \( j \) to node \( i \)). The graph is said to be symmetric if \( (j, i) \in E(t) \) implies \( (i, j) \in E(t) \). A graph is said to be connected if it is symmetric and if there is a path between any two vertices. A connected graph is complete if there is a direct connection between all vertices. An adjacency matrix for \( (V, E) \) is defined as an \( n \times n \) matrix \( G = [g_{ij}] \), where \( g_{ij} = 1 \) if \( w_{ij} > 0 \) and \( g_{ij} = 0 \) if \( w_{ij} = 0 \). A complete graph has a positive adjacency matrix. In this paper, we consider directed graphs, although more can be deduced for the special class of networks that can be represented with undirected graphs.

Example 1: Consider the four-node network depicted in Fig. 1. Let the averaging matrix be

\[
W = \begin{bmatrix}
1/2 & 1/2 & 0 & 0 \\
1/3 & 1/3 & 1/3 & 0 \\
0 & 1/3 & 1/3 & 1/3 \\
0 & 0 & 1/2 & 1/2
\end{bmatrix}.
\]

This network has the adjacency matrix

\[
G = \begin{bmatrix}
1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 \\
0 & 0 & 1 & 1
\end{bmatrix}
\]

which is symmetric and connected.

B. Stochastic Matrices

The value evolution can be described by the recursion

\[
x(t+1) = W(t)x(t), \quad W(t) \in \mathbb{W}
\]

where \( W(t) \) in the set \( \mathbb{W} \) is a nonnegative row-stochastic matrix with positive diagonal elements. For a row-stochastic matrix \( W \), all entries are nonnegative, and the rows add up to 1; therefore, we have \( We = e \) where \( e = [1, 1, \ldots, 1] \in \mathbb{R}^n \). In this paper, we are interested in the dynamic properties of (5) in the subspace \( \Delta = \{ \delta \in \mathbb{R}^n: \delta^T e = 0 \} \). Note that any vector in \( \mathbb{R}^n \) is in the joint span of \( \Delta \) and the vector \( e \). Hence, if we can demonstrate that the system (5) is stable in the
subspace $\Delta$, it would imply that the iterations of (5) would converge to a scalar multiple of $e$, which, in turn, means that all nodes have a common value.

**Lemma 1** (Markov [16]): Let y be a nonnegative vector and W a stochastic matrix. If $z = Wy$, then
\[
\max_i z_i - \min_i z_i \leq \tau(W) \left( \max_i y_i - \min_i y_i \right)
\]
where
\[
\tau(W) = \frac{1}{2} \max_{i,j} \sum_k |w_{ik} - w_{jk}|.
\]

The parameter $\tau(W)$ in (7) is referred to as the *coefficient of ergodicity* of $W$, and it satisfies $0 \leq \tau(W) \leq 1$ [16]. Furthermore, if $\tau(W) < 1$, the matrix $W$ is called a *scrambling* matrix. A scrambling matrix $W$ is contractive not only in $\Delta$, but also on the difference of the entries in a vector [16], [17]. A stochastic matrix $W$ is called *ergodic* if $\lim_{t \to \infty} W^t = ed^T$ for some $d$. A scrambling matrix is ergodic, but the converse is not true in general. This fact is illustrated with the following example.

**Example 2:** Reconsider the row-stochastic matrix in (3), for which the coefficient of ergodicity $\tau(W)$ can be computed to be $\tau(W) = 1$ from (7). Hence, this matrix is not scrambling. On the other hand, it is ergodic, as we have
\[
\lim_{t \to \infty} W^t = \begin{bmatrix} 0.2 & 0.3 & 0.3 & 0.2 \\ 0.2 & 0.3 & 0.3 & 0.2 \\ 0.2 & 0.3 & 0.3 & 0.2 \\ 0.2 & 0.3 & 0.3 & 0.2 \end{bmatrix} = ed^T
\]
where $d = [0.2, 0.3, 0.3, 0.2]^T$.

This paper relates the mathematical concepts of induced one norm, scrambling matrices, and ergodicity to nodes being synchronized. For instance, if there is a *master* node that distributes value information to others in the network, then this implies a scrambling matrix in our problem formulation. If such a master node exists all the time, it is extremely easy to note (both mathematically and practically) that the network value synchronization is achieved. We should note that one cannot rely on such centralized information in networks with time-varying topologies. The matrix in (3) is not a scrambling matrix (and there is no master node); however, synchronization is still achieved, since it is ergodic. Hence, the example matrix in (3) illustrates the fact that the scrambling matrix implies ergodicity, whereas ergodicity does not necessarily imply a scrambling matrix.

If all matrices in $\mathcal{W}$ are scrambling, then the matrix-induced one norm is common and decreasing for all matrices in the subspace $\Delta$; therefore we call it a *common Lyapunov function* (CLF) for the set $\mathcal{W}$. Given that such a CLF exists, it is straightforward to note that any trajectory in the subspace $\Delta$ is contracting; hence, consensus will be achieved under arbitrary switching. In the case that the existence of a CLF is not so obvious, it is important to relate the structural properties of the weighting matrices to synchronization, which is the focus of this paper in a probabilistic setting.

**Example 3:** To gain further insight into the problem, consider the following averaging matrices:
\[
W_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1/4 & 3/4 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]
\[
W_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 7/8 & 1/8 \end{bmatrix}
\]
\[
W_3 = \begin{bmatrix} 8/9 & 1/9 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]

which correspond to network topologies depicted in Figs. 2–4, respectively. It can be shown that neither matrix is ergodic. Also note that there is an isolated node in each graph.

Consider the situation in which the network topology is changing in a probabilistic fashion as shown in Figs. 2–4 with probabilities $p_1$, $p_2$, and $p_3$, respectively. The natural question that arises is under what conditions synchronization is achieved.

### III. Probabilistic Synchronization Algorithms

What lies at the heart of the results presented in [4]–[6] and [18]–[24] is that there is a node that affects all others at regular time intervals. Therefore, one needs to ensure connectedness of the network on the average to achieve synchronization. Although a centralized supervisor could be useful in some network configurations, this strategy might not be suitable for certain sensor network applications where access to the base station is always blocked, and hence, the necessity for distributed synchronization. In the sequel, we propose and study a probabilistic synchronization algorithm via averaging.

We consider a network in which the $j$th node transmits to the $i$th one with some fixed probability $\lambda_{ij}$. We assume an error-free communication medium that implies that the probability that the $i$th node receives information from the $j$th one is $\lambda_{ij}$. In this context, we propose that each processor updates its value according to (1) where
\[
w_{ij}(t) = \begin{cases} \frac{1}{1 + N_i(t)}, & \text{if node } i \text{ gets data from } j \\ 0, & \text{otherwise} \end{cases}
\]
and $N_i(t)$ is the number of transmissions that the $i$th node receives at time $t$. By using (11), we ensure that the received data are added to the
current value with equal weights, a heuristic that is called a random walk on a graph. Given the graph of a fixed network, one could optimize the mixing factor by appropriately choosing the averaging weights under certain cases [29]. However, in the absence of centralized information about the topology of the network coupled with the time-varying nature of the graph, random walk transitions would be adopted in the sequel.1

The value evolution can be described by (5) where

$$W(t) \in \mathcal{W} = \{W_1, W_2, \ldots, W_N\}, \quad N = 2^{(n^2-n)}.$$ 

Each $W_i$ is assumed to occur with some probability $\mu_j$ so that $\sum_{j=1}^{N} \mu_j = 1$. For $t > 0$, define the matrix products $X_t = \prod_{i=0}^{t} W(j)$. The problem under consideration is to determine conditions to assure almost sure consensus. A random agreement algorithm was considered in [28] where the authors studied random graphs with $\mu_j > 0$, an assumption that ensures that the graph of the network is complete almost surely. In this sense, the following result provides not only more relaxed conditions for almost sure consensus, but also an alternative proof for the result in [28].

**Theorem 1:** The probabilistic algorithm as described earlier achieves synchronization almost surely if and only if

$$\sum_{j=1}^{N} \mu_j W_j$$

is ergodic.

The proof of Theorem 1 relies on the following result by Berger [30].

**Lemma 2** (Berger [30]): Let $(W(j))$ be an independent and identically distributed sequence of $n \times n$ transition probability matrices. Assume that, for some $\epsilon > 0$, $\kappa = \epsilon < \infty$ almost surely, where

$$\kappa = \inf \{t > 0: \max_j \min_i (X_t)_{ij} \geq \epsilon \}.$$ 

Then, $X_t$ converges almost surely to a rank one matrix.

**Proof (Theorem 1):** Necessity is immediate. For the sufficiency part, suppose that $\overline{W} = \sum_{j=1}^{N} \mu_j W_j$ is ergodic. We will prove the result for a scrambling matrix $\overline{W}$; if $\overline{W}$ is not scrambling but ergodic, then $\overline{W}^{(k_0)}$ is scrambling for some $k_0 < n$, and hence, similar arguments could be repeated for $\overline{W}^{(k_0)}$. Since each matrix under consideration has positive diagonals, it is easy to note that the product $W_1 W_2 \cdots W_N$ is scrambling as well. Furthermore, from the fact that each matrix $W_i$ has a nonzero probability, it follows that the product $W_1 W_2 \cdots W_N$ must occur infinitely often (although the order of the matrices may differ). Hence, given $\epsilon > 0$, a finite $\kappa$ in Berger’s theorem (Lemma 2) exists almost surely, which implies that all products converge to a rank one matrix.

**Example 3 (continued):** Reconsider the arbitrary topologies in Example 3. It follows from Theorem 1 that synchronization is achieved if $p_i > 0$, $i = 1, 2, 3$.

**A. Convergence Rates**

Consider a network in which the probability that the $i$th node receives information from the $j$th one is $\lambda_{ij}$. In this section, we will investigate the relation between the convergence rate and the transmission probabilities. Denote $N = \{1, 2, \ldots, n\}$, $N_i = N - \{i\}$, $N_{ij} = N - \{i, j\}$; and $N_{[k]}^{(i)}, N_{ij}^{(k)}$ any subsets of $N_i$, and $N_{ij}$ of cardinality $k$, respectively. Taking the expectation of (1) and assuming that $W(t)$ is independent of $x(t)$, we have

$$E[x(t+1)] = E[W(t)x(t)] = E[W(t)]E[x(t)].$$

![Fig. 5. Simple network considered in Example 4.](image)

Let $\overline{x}(t) = E[x(t)]$ and $\overline{W} = E[W(t)]$, then the expected convergence rate of (1) is determined from that of the system

$$\overline{x}(t+1) = \overline{W}\overline{x}(t), \quad \overline{x}(0) = x(0)$$

where $\overline{W}$ is to be determined for the scenario under consideration. In our case, we assume that the $j$th node receives information from the $i$th one with probability $\lambda_{ij}$; therefore, we have the following result.

**Theorem 2:** The mean convergence rate of (1) can be approximated by that of (14) where the components of $\overline{W}$ are given by

$$\overline{W}_{i} = \sum_{k=0}^{n-1} \sum_{m \in N_i - N_{ij}^{(k)}} \lambda_{im} \prod_{l \in N_{ij}^{(k)}} (1 - \lambda_{lm})$$

for $i \neq j$

$$\overline{W}_{ij} = \lambda_{ij} \sum_{k=0}^{n-2} \sum_{m \in N_{ij} - N_{ij}^{(k)}} \lambda_{im} \prod_{l \in N_{ij}^{(k)}} (1 - \lambda_{lm}).$$

**Proof:** Consider the $i$th node, $w_{ii}$ may take values $1, 1/2, \ldots, 1/n$ with the probabilities given as follows.

<table>
<thead>
<tr>
<th>Value for $w_{ii}$ with probability</th>
<th>$1$</th>
<th>$1/2$</th>
<th>$1/3$</th>
<th>$\ldots$</th>
<th>$1/n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\prod_{m \in N_i} (1 - \lambda_{im})$</td>
<td>$\prod_{l \in N_{ij}^{(1)}} \lambda_{il} \prod_{m \in N_{ij} - N_{ij}^{(1)}} (1 - \lambda_{im})$</td>
<td>$\prod_{l \in N_{ij}^{(2)}} \lambda_{il} \prod_{m \in N_{ij} - N_{ij}^{(2)}} (1 - \lambda_{im})$</td>
<td>$\ldots$</td>
<td>$\prod_{l \in N_{ij}^{(n-1)}} \lambda_{il} \prod_{m \in N_{ij} - N_{ij}^{(n-1)}} (1 - \lambda_{im})$</td>
<td></td>
</tr>
</tbody>
</table>

Summing all expected values yields (15). $\overline{W}_{ij}$ can be computed similarly.

Given a maximum offset $d_0$ between the components of the initial state vector $x(0)$

$$d_0 = \max_i x_i(0) - \min_i x_i(0)$$

it is important to determine the number of steps $k_{max}$ required to achieve a desired level of synchronization accuracy $\epsilon$. From Lemma 1, $k_{max}$ can be computed as

$$k_{max} = \log_{\gamma(W)} (d_0 / \epsilon)$$

where $\gamma(W)$ is the expected matrix in Theorem 2. Note that $k_{max}$ obtained earlier is an upper bound.

**Example 4:** To illustrate the result in Theorem 2, consider the simple three-node network in Fig. 5, where the only data communication is from node 2 to nodes 1 and 3 with probabilities $\lambda_{12}$ and $\lambda_{32}$, respectively. In this case the expected matrix can be computed as

$$\overline{W} = \begin{bmatrix} 1 - \lambda_{12}/2 & \lambda_{12}/2 & 0 \\ 0 & 1 & 0 \\ 0 & \lambda_{32}/2 & 1 - \lambda_{32}/2 \end{bmatrix}$$

which is ergodic if and only if $\lambda_{12} > 0$ and $\lambda_{32} > 0$. 

1The proofs apply with any positive scalings of the transition matrices.
Theorem 2 delineates the mean convergence rate for general network topologies. Equations (15) and (16) and the lemma in the Appendix can be used to determine the convergence rates explicitly in some special cases.

Corollary 1: (i) For $\lambda_{ij} = \bar{\lambda}$, $\bar{W}$ has one eigenvalue at 1, and $n - 1$ eigenvalues at

$$1 - n\bar{\lambda} \sum_{k=0}^{n-2} \frac{1}{k+2} (n-2)_k (1-\bar{\lambda})^{n-k-2}.$$ (19)

(ii) If, at each instant, only a single node receives information from only one other with probability 1/2, then $\bar{W}$ is a doubly stochastic matrix with one eigenvalue at 1, and $(n-1)$ eigenvalues at

$$\frac{n-3/2}{n-1}.$$ (20)

B. Energy Considerations

Energy efficiency is critical in the operation of sensor networks, since battery life is limited, and sensing, collaboration, computation, and communication in the nodes all require energy. Incorporating energy awareness into the nodes by smart very-large-scale integration (VLSI) designs does not entirely solve the energy problem in sensor networks. The network as a whole should be energy aware, and in the sequel, we will examine such tradeoffs for the value synchronization problem by considering the energy spent only on transmissions (communication).

For the problem in hand, the average amount of communication from node $j$ to node $i$ is $\bar{\lambda}_{ij}$. Therefore, the total amount of energy spent on transmissions is easily computed as

$$E_{\text{average}} = \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} \bar{\lambda}_{ij},$$ (21)

A fundamental tradeoff can be observed between the connectivity of the network (which is determined by how full the matrix $\bar{W}$ is) and the convergence speed (related to the second largest eigenvalue of $\bar{W}$).

To this end, first consider the two extreme cases: 1) full connectivity ($\lambda_{ij} = 1, i \neq j$) in which case we have $\bar{W} = ee^T / n$ and 2) minimal connectivity where we assume that each node gets information from exactly one other node (e.g., Corollary 1(ii)).

When each node transmits to every other node at each instant (full connectivity), this results in consensus in one step (dead-beat system). Clearly, such a strategy requires a maximum amount of energy $E_{\text{average}} = n^2 - n$. On the other hand, in the case of minimal connectivity, the energy use is reduced substantially, although it also leads to slower convergence. For values of $\lambda_{ij} = \bar{\lambda} \in [0, 1]$, Fig. 6 clearly depicts the tradeoff between the average amount of transmissions [from (19)] and the magnitude of the second largest eigenvalue of the average matrix (from Corollary 1) that determines the convergence speed. For the synchronization problem, it is important to deduce the number of steps $k_{\text{max}}$ required to achieve a desired accuracy of synchronization, which is plotted in Fig. 7 for varying probability of transmission $\bar{\lambda}$ and $d_0/\epsilon = 0.001$. Not surprisingly, it is seen that synchronization can be sped up by increasing the transmission rate.

IV. CONCLUDING REMARKS

In this paper, we have studied the convergence properties of distributed probabilistic synchronization algorithms that can be used for communication networks. It is important to note the existence of a node (possibly different over different time intervals) distributing information to others so that the network could be synchronized. It is in this sense that the proposed probabilistic synchronization algorithm will find important applications in wireless sensor networks where the topology is time-varying, and connectivity may not be assured at all times.

APPENDIX

Lemma 3: Let $A = ce + \frac{1-c}{n} ee^T$ be an $n \times n$ matrix for some constant $c$. Then, its eigenvalues $\lambda_i$ and corresponding eigenvectors $v_i$ are

$$\lambda_1 = 1, \quad v_1 = e, \quad \lambda_2 = \cdots = \lambda_n = c, \quad v_2, \ldots, v_n \perp e.$$ (22)

Proof: For the first eigenvalue at 1, i.e., $\lambda_1 = 1$, we have

$$Av_1 = Ae = ce + \frac{1-c}{n} ee^T e = ce + (1-c)e = \lambda_1 v_1.$$
For the $i$th eigenvalue, $\lambda_i = 1, \ldots, n$, let $v_i \in \mathbb{R}^n$ be orthogonal to $e$, such that $e^T v_i = 0$. Then,

$$Av_i = cv_i + \frac{1 - c}{n} e^T v_i = cv_i, \quad i = 2, \ldots, n.$$ 

REFERENCES


