Information Dissemination in Networks via Linear Iterative Strategies Over Finite Fields

Shreyas Sundaram and Christoforos N. Hadjicostis

***Abstract*— Given an arbitrary network of interconnected nodes, each with an initial value from a discrete set, we consider the problem of distributively disseminating these initial values under the constraint that the nodes can only process and communicate values in that set. To solve this problem, we treat the initial values as elements in a finite field and employ a linear iterative strategy, whereby at each time-step, each node in the network transmits a value that is a linear combination of its own value and the most recent transmissions of its neighbors. Our approach is an important (and non-trivial) extension of previous work on linear iterative strategies with real-valued transmissions and operations, and can find applications in networks that have communication or computational constraints. We show that if the weights for the linear iteration are chosen randomly (uniformly and independently) from a finite field of sufficiently large size, then with some nonzero probability, any node will be able to obtain the initial value of any other node. Furthermore, this can be accomplished after running the linear iteration for a finite number of time-steps (upper bounded by the number of nodes in the network). In the process of deriving our results, we develop a theory of structured observability for linear systems over finite fields.**

I. Introduction

A key requirement in distributed systems and networks is to disseminate information from some or all of the nodes in the network to the other nodes. For example, sensor networks typically contain a set of nodes that sense some phenomenon in their environment; some or all of the nodes might then be required to calculate a certain function (such as the average) of these sensed values [1]. Various algorithms for information dissemination in networks have been developed by the computer science, communication, and control communities over the past few decades [2], [3], [4]. A particular strategy that has attracted significant attention in the control systems community is that of *linear iterations*; in this strategy, each node in the network repeatedly updates its value to be a weighted linear combination of its own value and those of its neighbors (e.g., see [5], [6] and the references therein). These works have revealed that if the network topology satisfies certain conditions, the weights for the linear iteration can be chosen to achieve objectives ranging from reaching asymptotic consensus (where all nodes in the network converge to the same function of the initial values), to calculating arbitrary functions of the initial values in a finite number of time-steps.

A common thread among the majority of these existing works on linear iterative strategies is that they consider networks with real-valued transmissions and operations. In many practical situations, however, networks will contain bandwidth restrictions in the transmission channels between nodes, or might contain nodes that are limited in the precision of the computations that they perform. With this constraint in mind, the topic of *quantized consensus* (where the values in the network are quantized to lie in some discrete set) has recently started to receive attention. For example, [7], [8], [9], [10] studied ways to obtain consensus by incorporating quantization into variants of a linear iterative strategy, where nodes update their values as a linear combination of quantized versions of their neighbors' values. An alternative method for quantized consensus based on *gossip* was proposed in [11], where each node periodically contacts a randomly chosen neighbor, and then they bring their values as close together as possible.

The problem of transmitting information (as opposed to reaching consensus) over finite fields has also been extensively studied by the communications community under the moniker of *network coding* [3], [12]. Much of the work in this area focuses on the topic of sending streams of information (instead of static initial values) from a set of source nodes to a set of sink nodes in the network. In this context, the prime concern is the maximum *rate* at which these streams can be transmitted, which is related to the network topology via the smallest cut between the source and sink nodes [3]. It has been shown that in certain networks, allowing intermediate nodes to mathematically combine the incoming source packets allows the streams to be transmitted at the maximum rate allowed by the network, whereas this cannot be achieved via simple routing schemes [12]. The extension of network coding to the problem of disseminating static initial values via a gossip algorithm has been investigated in [13], [14]. In these cases, each node periodically sends a random linear combination of the values that it currently holds to a randomly chosen neighbor. The analysis of such gossip protocols (for complete graphs in [13] and for more general graphs in [14]) is complex due to the probabilistic nature of the algorithm, but the authors of these

This material is based upon work supported in part by the National Science Foundation under NSF ITR Award 0426831 and NSF CNS Award 0834409. The research leading to these results has also received funding from the European Community Seventh Framework Programme (FP7/2007-2013) under grant agreements INFSO-ICT-223844 and PIRG02-GA-2007-224877. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the NSF or EC.

S. Sundaram is with the Coordinated Science Laboratory and the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, USA. E-mail: ssundarm@illinois.edu. C. N. Hadjicostis is with the Department of Electrical and Computer Engineering, University of Cyprus, and also with the Coordinated Science Laboratory, and the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign. E-mail: chadjic@ucy.ac.cy.

works provide bounds on the expected value of the number of gossip rounds required in order for all of the nodes to obtain all of the information.

In this paper, we study linear iterative strategies for disseminating static initial values (as was done in [6]), and extend such strategies to operate on values chosen from $\mathbb{F}_q$ (the finite field of size q). We show that these strategies can be effectively modeled as linear dynamical systems over finite fields, and then extend results from classical control theory to design and analyze such strategies. Our analysis produces an algorithm for disseminating information where multiple nodes in the network are allowed to simultaneously exchange information (rather than a gossip-based algorithm where only two random nodes exchange information at each round). In fixed strongly-connected networks where the initial values are elements of $\mathbb{F}_q$, we show that if the weights for the linear iteration are chosen randomly (independently and uniformly) from $\mathbb{F}_q$, then with high probability[1] (for sufficiently large q), each node can obtain the initial value of all other nodes after at most N time-steps (where N is the number of nodes in the network). This is in contrast to the work on quantized consensus and gossip-based network coding, where the expected convergence time can be much larger than the number of nodes in the network (as described above and in [9], [11], [13], [14]). Furthermore, while gossip-based network coding is capable of operating in unknown and potentially time-varying network topologies, this is achieved at the cost of increased communication and decoding complexity [13], [14]. In contrast, our work focuses on the problem of disseminating information in fixed time-invariant networks, and leverages this fact to provide a compact and systematic decoding strategy for each node in the network to recover the initial values.

Remark 1: It is instructive to compare the algorithm proposed in this paper (along with the gossip-based network coding algorithms in [13], [14]) to a simple "flooding" algorithm where each node sends any new information that it receives to all of its neighbors [2]. If there are no bandwidth constraints on the links in the network, then every node could simply send to its neighbors all of the information that it has at any point in time, and this would disseminate information within D time-steps, where D is the diameter is the network. On the other hand, if each node is only allowed to send one value from a given field at every time-step, then each node will have to carefully choose (or *schedule*) the values that it will transmit to its neighbors. Finding the best schedule for a given network is a difficult problem (e.g., see the discussion in [13]). Indeed, scheduling is a special case of the algorithms proposed here and in [13], [14], and thus these schemes will be able to do at least as well as scheduling (and strictly better in many cases, as shown in [13]). □

II. Notation and Background on Graph Theory

We use $\mathbf{e}_{i,l}$ to denote the column vector of length l with a "1" in its i-th position and "0" elsewhere. The symbol $\mathbf{1}_l$ denotes the column vector of length l with all entries equal to "1", and $\mathbf{I}_N$ denotes the $N \times N$ identity matrix. We will also denote the cardinality of a set $\mathcal{S}$ by $|\mathcal{S}|$, and use the notation $\text{diag}\,(\cdot)$ to indicate a square matrix with the quantities inside the brackets on the diagonal, and zeros elsewhere. The transpose of a matrix $\mathbf{A}$ will be denoted by $\mathbf{A}'$. The probability of an event A is denoted by $\Pr[A]$.

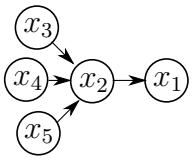

Fig. 1. Spanning tree rooted at x_1.

A graph is an ordered pair $\mathcal{G} = \{\mathcal{X}, \mathcal{E}\}$, where $\mathcal{X} = \{x_1, \ldots, x_N\}$ is a set of vertices, and $\mathcal{E}$ is a set of ordered pairs of different vertices, called directed edges. If $(x_i, x_j) \in \mathcal{E} \Leftrightarrow (x_j, x_i) \in \mathcal{E}$, the graph is said to be undirected. The nodes in the set $\mathcal{N}_i = \{x_j | (x_j, x_i) \in \mathcal{E}\}$ are said to be neighbors of node x_i, and the in-degree of node x_i is denoted by $\deg_i = |\mathcal{N}_i|$. Similarly, the out-degree of node x_i is the number of nodes that have x_i as a neighbor. A *subgraph* of $\mathcal{G}$ is a graph $\mathcal{H} = \{\bar{\mathcal{X}}, \bar{\mathcal{E}}\}$, with $\bar{\mathcal{X}} \subseteq \mathcal{X}$ and $\bar{\mathcal{E}} \subseteq \mathcal{E}$ (where all edges in $\bar{\mathcal{E}}$ are between vertices in $\bar{\mathcal{X}}$).

A *path* P from vertex x_{i_0} to vertex x_{i_t} is a sequence of vertices $x_{i_0}, x_{i_1}, \ldots, x_{i_t}$ such that $(x_{i_j}, x_{i_{j+1}}) \in \mathcal{E}$ for $0 \leq j \leq t-1$. A path is called a *cycle* if its start vertex and end vertex are the same, and no other vertex appears more than once in the path. A graph $\mathcal{G}$ is a *spanning tree rooted at* x_i if it is an acyclic graph where every node has a path to x_i, and every node except x_i has out-degree exactly equal to 1. An example of a spanning tree rooted at x_1 is shown in Fig. 1. A graph is *strongly-connected* if there is a path from every node to every other node. Further background on graph theory can be found in standard texts, such as [15].

III. Problem Formulation

Consider a network modeled by the directed graph $\mathcal{G} = \{\mathcal{X}, \mathcal{E}\}$, where $\mathcal{X} = \{x_1, \ldots, x_N\}$ is the set of N nodes and directed edge $(x_j, x_i) \in \mathcal{E}$ if node x_i can receive information directly from node x_j. Each node x_i has some initial value $x_i[0]$ (from some field $\mathbb{F}$) that is potentially required by other nodes in order to calculate certain functions. We study a linear iterative strategy to disseminate these values through the network; specifically, at each time-step k, each node updates its value as

$$x_i[k+1] = w_{ii} x_i[k] + \sum_{j \in \mathcal{N}_i} w_{ij} x_j[k],$$

where the w_{ij}'s are a set of weights from $\mathbb{F}$ and all operations are performed over that field. For ease of analysis, the values of all nodes at time-step k can be aggregated into the value vector $\mathbf{x}[k] = \begin{bmatrix} x_1[k] & x_2[k] & \cdots & x_N[k] \end{bmatrix}'$, so that the

[1]For this (high) probability, we establish a lower bound that increases with the size q of the underlying finite field.

update strategy for the entire system can be represented as

$$\mathbf{x}[k+1] = \underbrace{\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1N} \\ w_{21} & w_{22} & \cdots & w_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N1} & w_{N2} & \cdots & w_{NN} \end{bmatrix}}_{\mathbf{W}} \mathbf{x}[k] \qquad (1)$$

for $k = 0, 1, \ldots$, with the constraint that $w_{ij} = 0$ if $j \notin \mathcal{N}_i$. Based on the above strategy, we study the following problem.

Problem: Characterize conditions on the network topology and find weights in a finite field $\mathbb{F}_q$ (of size q) so that every node in the (time-invariant) network can obtain the initial values of all other nodes after using a linear iterative strategy, where all operations and transmissions involve only elements of $\mathbb{F}_q$.

To solve this problem, we first need to model the information that each node receives via the linear iterative strategy. Let $\mathbf{y}_i[k]$ denote the vector of outputs (node values) that node x_i receives at the k–th time-step. Specifically, since node x_i has access to its own value as well as the values of its neighbors, we can write

$$\mathbf{y}_i[k] = \mathbf{C}_i \mathbf{x}[k], \;\; 1 \leq i \leq N \;\; , \qquad (2)$$

where $\mathbf{C}_i$ is the $(\deg_i + 1) \times N$ matrix with a single "1" in each row denoting the positions of the state-vector $\mathbf{x}[k]$ that are available to node x_i (i.e., these positions correspond to the nodes that are neighbors of node x_i, along with node x_i itself). Since $\mathbf{x}[k] = \mathbf{W}^k \mathbf{x}[0]$, the set of all outputs seen by node x_i over $L+1$ time-steps is given by

$$\underbrace{\begin{bmatrix} \mathbf{y}_i[0] \\ \mathbf{y}_i[1] \\ \mathbf{y}_i[2] \\ \vdots \\ \mathbf{y}_i[L] \end{bmatrix}}_{\mathbf{y}_i[0:L]} = \underbrace{\begin{bmatrix} \mathbf{C}_i \\ \mathbf{C}_i\mathbf{W} \\ \mathbf{C}_i\mathbf{W}^2 \\ \vdots \\ \mathbf{C}_i\mathbf{W}^L \end{bmatrix}}_{\mathcal{O}_{i,L}} \mathbf{x}[0] \;\; . \qquad (3)$$

When $L = N-1$, the matrix $\mathcal{O}_{i,L}$ in the above equation is the *observability matrix* for the pair $(\mathbf{W}, \mathbf{C}_i)$ [16]; in this paper, we will use the term *observability matrix* to refer to $\mathcal{O}_{i,L}$ for any L. If the row space of the observability matrix $\mathcal{O}_{i,L}$ contains a matrix $\mathbf{Q}$, then one can find a matrix Γ_i such that $\Gamma_i \mathcal{O}_{i,L} = \mathbf{Q}$. Thus, after running the linear iteration (1) for $L+1$ time-steps, node x_i can calculate $\mathbf{Q}\mathbf{x}[0]$ as

$$\Gamma_i \mathbf{y}_i[0:L] = \Gamma_i \mathcal{O}_{i,L} \mathbf{x}[0] = \mathbf{Q}\mathbf{x}[0] \;\; . \qquad (4)$$

If $\text{rank}(\mathcal{O}_{i,N-1}) = N$, the pair $(\mathbf{W}, \mathbf{C}_i)$ is said to be *observable*. In this case, node x_i can determine the entire initial value vector $\mathbf{x}[0]$ from the outputs of the system, and can therefore calculate any function of those values.

The approach described above was used in [6] to analyze linear iterative strategies over the field of complex numbers (with real-valued transmissions and operations). In that case, it was shown that for almost any[2] real-valued choice of weights, the nodes in the system can calculate an arbitrary function of the other node values after running the linear iteration (1) for a finite number of time-steps (as long as there are paths from the nodes that hold the needed values to the nodes that have to calculate the functions). This result was derived in [6] by appealing to a branch of control theory pertaining to *linear structured systems*. However, this analysis technique does not directly apply to finite fields because the existing literature on structured systems only deals with the field of complex numbers. In order to extend linear iterative strategies to finite fields, we will start in the next section by developing a theory of structured system observability over finite fields.

Remark 2: It was shown in [6] that the nodes in the network do not need to know the entire network topology or weight matrix *a priori* in order to use the above strategy. Instead, the nodes in the (time-invariant) network can discover the required information about the network after following a simple distributed protocol. Along similar lines, each node in the network does not need to store the entire set of values $\mathbf{y}_i[0:L]$ in order to calculate the function via equation (4). If r denotes the number of rows in $\mathbf{Q}$, then one can show that each node x_i only requires r additional registers in order to calculate $\mathbf{Q}\mathbf{x}[0]$. The interested reader is referred to [6] for more details; while that paper deals with linear iterative strategies with real-valued transmissions and operations, the ideas described above carry over to finite fields as well. □

IV. Observability of Linear Systems over Finite Fields

In order to characterize the observability of the pair $(\mathbf{W}, \mathbf{C}_i)$ for arbitrary networks when the weights for the linear iteration are chosen from a finite field, we will start by investigating the observability of matrix pairs of the form $(\mathbf{A}, \mathbf{e}'_{1,N})$, where $\mathbf{A}$ is an $N \times N$ matrix, and $\mathbf{e}'_{1,N}$ is a row-vector of length N with a 1 in its first position and zeros elsewhere. Matrix $\mathbf{A}$ is said to be *structured* if every entry of $\mathbf{A}$ is either zero, or an independent free parameter (to be chosen from a field $\mathbb{F}$). Our analysis will be based on a graph representation of matrix $\mathbf{A}$, denoted by $\mathcal{H}$, which we obtain as follows. The vertex set of $\mathcal{H}$ is $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$, and the edge set is given by $\mathcal{E} = \{(x_j, x_i) \mid \mathbf{A}_{ij} \neq 0\}$. The weight on edge (x_j, x_i) is set to the value of $\mathbf{A}_{ij}$ (this can be a free parameter if $\mathbf{A}$ a structured matrix). Note that if $\mathbf{A}$ is the weight matrix for a linear iteration, $\mathcal{H}$ is simply the graph of the network $\mathcal{G}$, augmented with self-loops on every node. Graph-based characterizations of properties such as observability have previously been investigated under the moniker of *structural system theory* [17]; however these works assume that the field under consideration is the field of complex numbers, where the parameters of the system matrices are allowed to take on any real values. In this section, we will develop a graph-based theory of observability of linear systems over arbitrary fields, and then apply our results to the topic of information dissemination over finite fields.

[2]The phrase "almost any" in this context means that the set of weights for which the property is violated has Lebesgue measure zero.

A. Observability of A Spanning Tree

We will start with the following result, which considers a linear system whose graph is a spanning tree.

Theorem 1: Consider the matrix pair $(\mathbf{A}, \mathbf{e}'_{1,N})$, where the entries in $\mathbf{A}$ are elements of a field $\mathbb{F}$ with size at least N. Suppose that the following two conditions hold:

- The graph $\mathcal{H}$ associated with $\mathbf{A}$ is a spanning tree rooted at x_1, augmented with self-loops on every node.
- The weights on the self-loops are different elements of $\mathbb{F}$ for every node, and the weights on the edges between different nodes are equal to 1.

Then the pair $(\mathbf{A}, \mathbf{e}'_{1,N})$ is observable over the field $\mathbb{F}$. □

Proof: Since the graph associated with $\mathbf{A}$ is a spanning tree rooted at x_1, there exists a numbering[3] of the nodes such that the $\mathbf{A}$ matrix is upper-triangular, with the self-loop weights on the diagonal [17], [15]. Denote the self-loop weight on node x_i by λ_i. Since all of the self-loop weights are different, this matrix will have N distinct eigenvalues (given by $\lambda_1, \lambda_2, \ldots, \lambda_N$), with N corresponding linearly independent eigenvectors.

Consider the eigenvalue λ_i. Let x_l be any node in the graph with in-degree 0 such that the path from x_l to x_1 passes through x_i (if the in-degree of x_i is zero, we can take $x_l = x_i$). Let r_i denote the number of nodes in this path, and renumber the nodes (leaving x_1 unchanged) so that all nodes on the path from x_l to x_1 come first in the ordering, and all other nodes come next. Let $\mathbf{P}_i$ denote the permutation matrix that corresponds to this reordering, and note that the matrix $\mathbf{P}_i \mathbf{A} \mathbf{P}_i^{-1}$ has the form

$$\mathbf{P}_i \mathbf{A} \mathbf{P}_i^{-1} = \begin{bmatrix} \mathbf{J}_i & \bar{\mathbf{A}}_1 \\ \mathbf{0} & \bar{\mathbf{A}}_2 \end{bmatrix} \quad , \tag{5}$$

for some matrices $\bar{\mathbf{A}}_1$ and $\bar{\mathbf{A}}_2$. The matrix $\mathbf{J}_i$ has the parameters $\lambda_1, \lambda_2, \ldots, \lambda_{r_i}$ on the diagonal (where each of these parameters is a different element of $\mathbb{F}$), each entry on the superdiagonal is "1", and all other entries are "0". This matrix has r_i distinct eigenvalues (given by the λ_t's) in the field $\mathbb{F}$, and thus the matrix has r_i eigenvectors over $\mathbb{F}$. Note that there exists some $t \in \{1, 2, \ldots, r_i\}$ such that $\lambda_t = \lambda_i$ (where λ_i is the eigenvalue that we are considering in matrix $\mathbf{A}$). It is easy to verify that the eigenvector $\mathbf{v}_t$ of $\mathbf{J}_i$ associated with the eigenvalue λ_t is given by

$$\mathbf{v}_t = \begin{bmatrix} 1 & (\lambda_t - \lambda_1) & (\lambda_t - \lambda_1)(\lambda_t - \lambda_2) & \cdots \\ & \cdots & \prod_{s=1}^{t-1}(\lambda_t - \lambda_1) & 0 & \cdots & 0 \end{bmatrix}' \quad ,$$

and thus the eigenvector corresponding to eigenvalue λ_t for the matrix $\mathbf{P}_i \mathbf{A} \mathbf{P}_i^{-1}$ in equation (5) is given by $\mathbf{w}_t = \begin{bmatrix} \mathbf{v}_t \\ \mathbf{0} \end{bmatrix}$. Next, note that the eigenvector corresponding to eigenvalue λ_t (or equivalently, λ_i) for matrix $\mathbf{A}$ will be given by $\mathbf{P}_i \mathbf{w}_t$. Since $\mathbf{P}_i$ is a permutation matrix, and node x_1 was left unchanged during the permutation, the first row of $\mathbf{P}_i$ is given by the vector $\mathbf{e}'_{1,N}$. This means that the first element of the eigenvector $\mathbf{P}_i \mathbf{w}_t$ will be "1" (based on the matrices $\mathbf{w}_t$ and $\mathbf{v}_t$ shown above). Since the above analysis holds for any eigenvalue λ_i, we can conclude that the eigenvectors associated with all eigenvalues for the matrix $\mathbf{A}$ will have a "1" as their first element. Let $\mathbf{V}$ be the matrix whose columns contain these eigenvectors (so that each entry in the first row of $\mathbf{V}$ is "1"). We thus have $\mathbf{V}^{-1} \mathbf{A} \mathbf{V} = \Lambda$, where $\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_N)$, and furthermore, $\mathbf{e}'_{1,N} \mathbf{V} = \mathbf{1}'_N$. Since this is a similarity transformation, the rank of the observability matrix for the pair $(\mathbf{A}, \mathbf{e}'_{1,N})$ will be the same as the rank of the observability matrix for the pair $(\Lambda, \mathbf{1}'_N)$. The observability matrix for the pair $(\Lambda, \mathbf{1}'_N)$ has the form of a *Vandermonde matrix* in the parameters $\lambda_1, \lambda_2, \ldots, \lambda_N$ [18]. It is well-known that such matrices are invertible if and only if all of the parameters are distinct, and thus the above observability matrix has rank N over $\mathbb{F}$ [18]. This means that the pair $(\mathbf{A}, \mathbf{e}'_{1,N})$ will also be observable. ■

[3]This renumbering corresponds to a similarity transformation on $\mathbf{A}$ with a permutation matrix, and does not change the eigenvalues of $\mathbf{A}$.

B. Observability of Arbitrary Graphs

We will now consider matrices $\mathbf{A}$ with arbitrary graphs. The following corollary is immediate from the previous section.

Corollary 1: Consider the matrix pair $(\mathbf{A}, \mathbf{e}'_{1,N})$, where $\mathbf{A}$ is a structured matrix (i.e., every entry of $\mathbf{A}$ is either a fixed zero or an independent free parameter from a field $\mathbb{F}$). Suppose the graph $\mathcal{H}$ associated with the matrix $\mathbf{A}$ contains a path from every node to node x_1, and furthermore, every node has a self-loop (i.e., the diagonal elements of $\mathbf{A}$ are free parameters). Then, if $\mathbb{F}$ has size at least N, there exists a choice of parameters from $\mathbb{F}$ such that the observability matrix for the pair $(\mathbf{A}, \mathbf{e}'_{1,N})$ has rank N over $\mathbb{F}$.

Proof: Since there is a path from every node to node x_1 in $\mathcal{H}$, it is possible to find a subgraph of $\mathcal{H}$ that is a spanning tree rooted at x_1 (this subgraph contains every node of $\mathcal{H}$, but not every edge). Let $\bar{\mathcal{H}}$ be this subgraph augmented with self-loops on every node. Now, set the values of all parameters corresponding to edges that are not in $\bar{\mathcal{H}}$ to zero, and set the values of all parameters corresponding to edges between different nodes in $\bar{\mathcal{H}}$ to 1. Finally, set all of the parameters corresponding to the self-loops to be *different* values from the field $\mathbb{F}$ (this is possible since the size of the field is at least N). This produces a matrix $\mathbf{A}$ satisfying the conditions in Theorem 1, and thus the pair $(\mathbf{A}, \mathbf{e}'_{1,N})$ is observable with this choice of parameters. ■

The above corollary shows that one can explicitly choose parameters from a field of size N or greater in order to make the pair $(\mathbf{A}, \mathbf{e}'_{1,N})$ observable. However, we will also be interested in the case where each nonzero parameter in $\mathbf{A}$ is chosen *randomly* (i.e., independently and uniformly) from $\mathbb{F}_q$ (the finite field of size q).

Theorem 2: Consider the matrix pair $(\mathbf{A}, \mathbf{e}'_{1,N})$, where $\mathbf{A}$ is a structured matrix (i.e., every entry of $\mathbf{A}$ is either a fixed zero or an independent free parameter from field $\mathbb{F}_q$ of size $q \geq \frac{N(N-1)}{2}$). Suppose the graph $\mathcal{H}$ associated with the matrix $\mathbf{A}$ contains a path from every node to node x_1, and furthermore, every node has a self-loop (i.e., the diagonal elements of $\mathbf{A}$ are free parameters). Then, if each

free parameter in $\mathbf{A}$ is chosen randomly (independently and uniformly) from the field $\mathbb{F}_q$, the probability that the pair $(\mathbf{A}, \mathbf{e}'_{1,N})$ will be observable is at least $1 - \frac{N(N-1)}{2q}$. □

To prove this theorem, we will make use of the following lemma (e.g., see [19], [20]). In this lemma, the *total degree* of a multivariate polynomial $p(\xi_1, \xi_2, \ldots, \xi_n)$ is defined as the maximum sum of the degrees of the variables $\xi_1, \xi_2, \ldots, \xi_n$ in any term of the polynomial.

Lemma 1 (Schwartz-Zippel): Let $p(\xi_1, \xi_2, \ldots, \xi_n)$ be a nonzero polynomial of total degree d with coefficients in finite field $\mathbb{F}_q$ (with $q \geq d$). If p is evaluated on an element $(s_1, s_2, \ldots, s_n)$ chosen uniformly and independently from $\mathbb{F}_q^n$, then $\Pr[p(s_1, s_2, \ldots, s_n) = 0] \leq \frac{d}{q}$.

We will now prove Theorem 2.

Proof: Let the free parameters of matrix $\mathbf{A}$ be given by $\lambda_1, \lambda_2, \ldots, \lambda_l \in \mathbb{F}_q$ (note that these λ_i's no longer refer to only the diagonal entries, as in the proof of Theorem 1, but to all nonzero entries in the matrix). When convenient, we will aggregate these parameters into a vector $\lambda \in \mathbb{F}_q^l$, and denote the matrix $\mathbf{A}$ by $\mathbf{A}(\lambda)$ to explicitly show its dependence on the free parameters. Any particular choice of the free parameters will be denoted by $\lambda^* = \begin{bmatrix}\lambda_1^* & \lambda_2^* & \cdots & \lambda_l^*\end{bmatrix}$, with corresponding numerical matrix $\mathbf{A}(\lambda^*)$.

The observability matrix $\mathcal{O}(\lambda)$ for the pair $(\mathbf{A}(\lambda), \mathbf{e}'_{1,N})$ has the form

$$\mathcal{O}(\lambda) = \begin{bmatrix} \mathbf{e}'_{1,N} \\ \mathbf{e}'_{1,N}\mathbf{A}(\lambda) \\ \vdots \\ \mathbf{e}'_{1,N}\mathbf{A}^{N-1}(\lambda) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ * & \psi_{1,2} & \psi_{1,3} & \cdots & \psi_{1,N} \\ * & \psi_{2,2} & \psi_{2,3} & \cdots & \psi_{2,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ * & \psi_{N-1,2} & \psi_{N-1,3} & \cdots & \psi_{N-1,N} \end{bmatrix},$$

where $*$ represents unimportant quantities, and each $\psi_{i,j}$ is a polynomial in the free parameters of matrix $\mathbf{A}$. Recall that entry $(1, j)$ in $\mathbf{A}^i(\lambda)$ is a polynomial where every term is a product of the weights on a path of length i from node x_j to node x_1 (i.e., each term will be product of i free parameters, where some of the parameters might be repeated) [15]. Thus, every term in the polynomial $\psi_{i,j}$ will be the product of i free parameters (duplications included). The determinant of matrix $\mathcal{O}(\lambda)$ in terms of the quantities $\psi_{i,j}$ is given by

$$\det \mathcal{O}(\psi_{1,2}, \psi_{1,3}, \ldots, \psi_{N-1,N}) = \sum_{\tau_{i_{1,2}} + \cdots + \tau_{i_{N-1,N}} = N-1} c_{i_{1,2},\ldots,i_{N-1,N}} \psi_{1,2}^{\tau_{i_{1,2}}} \cdots \psi_{N-1,N}^{\tau_{i_{N-1,N}}},$$

where each $\tau_{i_{jt}} \in \{0, 1\}$ and $c_{i_{1,2},i_{1,3},\ldots,i_{N-1,N}} \in \{-1, 0, 1\}$ for each choice of $\tau_{i_{1,2}}, \tau_{i_{1,3}}, \ldots, \tau_{i_{N-1,N}}$ satisfying the above constraints; furthermore, each term will contain parameters $\psi_{i,j}$ from different rows and columns of $\mathcal{O}(\psi_{1,2}, \psi_{1,3}, \ldots, \psi_{N-1,N})$ [17]. Since each term in $\psi_{i,j}$ will contribute i parameters from λ, we see that every term in $\det \mathcal{O}(\psi_{1,2}, \ldots, \psi_{N-1,N})$ can be expanded to a sum of terms, each of which is a product of no more than $1 + 2 + 3 + \cdots + N - 1 = \frac{N(N-1)}{2}$ parameters from λ (duplications included). More precisely, we can write

$$\det \mathcal{O}(\lambda) = \sum_{\eta_{i_1} + \cdots + \eta_{i_l} \leq \frac{N(N-1)}{2}} d_{i_1,\ldots,i_l} \lambda_1^{\eta_{i_1}} \cdots \lambda_l^{\eta_{i_l}},$$

where $\eta_{i_j} \geq 0$ for $j \in \{1, 2, \ldots, l\}$, and $d_{i_1,i_2,\ldots,i_l}$ is some constant for each term in the polynomial. This is a polynomial of total degree no greater than $\frac{N(N-1)}{2}$.

Now, note from Corollary 1 that if the graph of matrix $\mathbf{A}$ satisfies the conditions in the theorem, then there exists a choice of parameters $\lambda^* \in \mathbb{F}_q^l$ (where $q \geq N$) such that the pair $(\mathbf{A}(\lambda^*), \mathbf{e}'_{1,N})$ is observable over the field $\mathbb{F}_q$. This means that $\det \mathcal{O}(\lambda)$ is a nonzero polynomial over $\mathbb{F}_q$ (since it is nonzero for the *particular* choice $\lambda = \lambda^*$). If we now choose a set of parameters $(\lambda_1^*, \lambda_2^*, \ldots, \lambda_l^*)$ independently and uniformly from $\mathbb{F}_q^l$, we can apply the Schwartz-Zippel lemma (Lemma 1) to obtain $\Pr[\det \mathcal{O}(\lambda^*) = 0] \leq \frac{N(N-1)}{2q}$, or equivalently, $\Pr[\det \mathcal{O}(\lambda^*) \neq 0] \geq 1 - \frac{N(N-1)}{2q}$. This concludes the proof of the theorem. ■

V. Linear Iterative Strategies over Finite Fields

The linear iterative strategy given by equations (1) and (2) essentially defines a structured linear system (i.e., each entry of the matrix $\mathbf{W}$ is either identically zero, or an independent free parameter). The following theorem describes how to choose the free parameters (i.e., the weights for the linear iteration) so that the pair $(\mathbf{W}, \mathbf{C}_i)$ will be observable for every i.

Theorem 3: Let $\mathcal{G} = \{\mathcal{X}, \mathcal{E}\}$ denote the graph of a strongly-connected network. If the initial values of the nodes are elements of a field $\mathbb{F}_q$ (of size $q \geq \frac{N^2(N-1)}{2}$) and if the weights for the linear iteration are chosen independently and uniformly from that field, then with probability at least $1 - \frac{N^2(N-1)}{2q}$, every node x_i can obtain the initial values of all other nodes after running the linear iteration (1) for $L_i + 1$ time-steps, for some $0 \leq L_i < N - 1$. □

Proof: Consider the observability matrix for the pair $(\mathbf{W}, \mathbf{C}_i)$ for any node x_i in the network. Without loss of generality, suppose that $i = 1$ (the nodes can simply be renumbered so that this is the case). Let $\mathcal{H}$ denote the graph associated with $\mathbf{W}$ (note that this is simply the graph of the network, augmented with self-loops on every node). Since the network is strongly-connected, there is a path from any node to x_1 in $\mathcal{H}$. Theorem 2 indicates that if the weights are chosen independently and uniformly from $\mathbb{F}_q$, the observability matrix for the pair $(\mathbf{W}, \mathbf{e}'_{1,N})$ will have rank N with probability at least $1 - \frac{N(N-1)}{2q}$. Since the observability matrix for the pair $(\mathbf{W}, \mathbf{e}'_{1,N})$ is a submatrix of the observability matrix for the pair $(\mathbf{W}, \mathbf{C}_1)$, the pair $(\mathbf{W}, \mathbf{C}_1)$ will also be observable with probability at least $1 - \frac{N(N-1)}{2q}$.

Now, note that the above analysis holds for any node $x_i \in \mathcal{X}$. Let $\mathbf{W}$ be a weight matrix with weights chosen at random from $\mathbb{F}_q$, and denote the observability matrix for node x_i by $\mathcal{O}_i$. Using the union bound, the probability that every node

$x_i \in \mathcal{X}$ can obtain the initial values of all other nodes in the network satisfies

$$\Pr\left[\bigcap_{x_i \in \mathcal{X}} \text{rank}(\mathcal{O}_i) = N\right] \geq 1 - \frac{N^2(N-1)}{2q} \ .$$

■

Remark 3: Note that the probability bound obtained in the above theorem is potentially quite loose because of several conservative assumptions in our derivation, such as the union bound and the Schwartz-Zippel Lemma. It may the case that one can use finite fields of much smaller size than that specified by the above theorem in order to ensure that the observability matrix for every node will be of full column rank. The main contribution of the above result is to show that there always exists a finite field of sufficiently large size over which linear iterative strategies can be applied (in strongly-connected networks). □

VI. Example

Consider an undirected ring network with $N = 5$ nodes, with connections $x_1 \leftrightarrow x_2 \leftrightarrow x_3 \leftrightarrow x_4 \leftrightarrow x_5 \leftrightarrow x_1$. Since this network is strongly-connected, Theorem 3 indicates that choosing weights from a field of size $q \geq \frac{N^2(N-1)}{2} = 50$ will ensure that the observability matrices for all nodes will be full rank with probability at least $1 - \frac{50}{q}$. For instance, if we choose $q = 1023$, we obtain a probability of success at least 95.1%. Choosing the weights independently and uniformly from the field $\mathbb{F}_{1023}$ we obtain the weight matrix

$$\mathbf{W} = \begin{bmatrix} 282 & 509 & 0 & 0 & 981 \\ 695 & 981 & 260 & 0 & 0 \\ 0 & 348 & 517 & 833 & 0 \\ 0 & 0 & 715 & 152 & 249 \\ 121 & 0 & 0 & 263 & 950 \end{bmatrix} \ .$$

Consider node x_2 in the network; the values that this node receives at each time-step are given by $\mathbf{y}_2[k] = \mathbf{C}_2\mathbf{x}[k] = \begin{bmatrix}\mathbf{I}_3 & \mathbf{0}\end{bmatrix}$. The observability matrix after two time-steps has the form $\mathcal{O}_{2,2} = \begin{bmatrix}\mathbf{C}_2 \\ \mathbf{C}_2\mathbf{W}\end{bmatrix}$ which has rank 5 over the field $\mathbb{F}_{1023}$. Thus, we can find a matrix Γ_2 satisfying $\Gamma_2\mathcal{O}_{2,2} = \mathbf{I}_5$ and provide this matrix to node x_2. The procedure can be repeated for all other nodes in the network.

Now suppose that the initial values of the nodes are $\mathbf{x}[0] = \begin{bmatrix}191 & 246 & 1001 & 0 & 598\end{bmatrix}'$. After one iteration, the values of the nodes are $\mathbf{x}[1] = \mathbf{W}\mathbf{x}[0] = \begin{bmatrix}510 & 71 & 578 & 182 & 940\end{bmatrix}'$ (note that all operations are performed in the field $\mathbb{F}_{1023}$, i.e., multiplication and addition are taken modulo 1023). The values seen by node x_2 over these two time-steps are $\mathbf{y}_2[0] = \mathbf{C}_2\mathbf{x}[0] = \begin{bmatrix}191 & 246 & 1001\end{bmatrix}'$ and $\mathbf{y}_2[1] = \mathbf{C}_2\mathbf{x}[1] = \begin{bmatrix}510 & 71 & 578\end{bmatrix}'$. Node x_2 can now obtain the entire set of initial values as $\mathbf{x}[0] = \Gamma_2\begin{bmatrix}\mathbf{y}_2'[0] & \mathbf{y}_2'[1]\end{bmatrix}'$, and can therefore calculate any desired function of those values. It can be verified that the same holds true for any other node in the network, which means that the nodes can also reach consensus (if desired) on any function of the initial values after two time-steps.

VII. Summary

In this paper, we considered the problem of using linear iterative strategies to disseminate information in networks where operations are performed in a finite field $\mathbb{F}_q$. We obtained a lower bound on the probability that linear iterative strategies will allow nodes in arbitrary networks to obtain the initial values of all other nodes if the weights are chosen uniformly and independently from a field of size q. To do this, we developed a theory of structured system observability over finite fields.

References

[1] C. G. Cassandras and W. Li, "Sensor networks and cooperative control," *European Journal of Control*, vol. 11, no. 4–5, pp. 436–463, 2005.

[2] N. A. Lynch, *Distributed Algorithms*. Morgan Kaufmann Publishers, Inc., 1996.

[3] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 782–795, Oct. 2003.

[4] A. Giridhar and P. R. Kumar, "Toward a theory of in-network computation in wireless sensor networks," *IEEE Communications Magazine*, vol. 44, no. 4, pp. 98–107, Apr. 2006.

[5] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[6] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation and consensus using linear iterative strategies," *IEEE Journal on Selected Areas in Communications: Special Issue on Control and Communications*, vol. 26, no. 4, pp. 650–660, May 2008.

[7] M. E. Yildiz and A. Scaglione, "Differential nested lattice encoding for consensus problems," in *Proceedings of the 6th International Conference on Information Processing in Sensor Networks (IPSN)*, 2007, pp. 89–98.

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

[9] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "On distributed averaging algorithms and quantization effects," in *Proceedings of the 47th IEEE Conference on Decision and Control*, 2008, pp. 4825–4830.

[10] P. Frasca, R. Carli, F. Fagnani, and S. Zampieri, "Average consensus by gossip algorithms with quantized communication," in *Proceedings of the 47th IEEE Conference on Decision and Control*, 2008, pp. 4837–4842.

[11] A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192–1203, July 2007.

[12] R. W. Yeung, S.-Y. R. Li, N. Cai, and Z. Zhang, "Network coding theory," in *Foundations and Trends in Communications and Information Theory*. now Publishers Inc., 2005, vol. 2, no. 4/5, pp. 241–381.

[13] S. Deb, M. Médard, and C. Choute, "Algebraic gossip: A network coding approach to optimal multiple rumor mongering," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2486–2507, June 2006.

[14] D. Mosk-Aoyama and D. Shah, "Information dissemination via network coding," in *Proceedings of the 2006 IEEE International Symposium on Information Theory*, 2006, pp. 1748–1752.

[15] D. B. West, *Introduction to Graph Theory*. Prentice-Hall Inc., Upper Saddle River, New Jersey, 2001.

[16] C.-T. Chen, *Linear System Theory and Design*. Holt, Rinehart and Winston, 1984.

[17] K. J. Reinschke, *Multivariable Control A Graph-Theoretic Approach*. Springer-Verlag, 1988.

[18] R. E. Blahut, *Algebraic Codes for Data Transmission*. Cambridge University Press, 2003.

[19] T. Ho, M. Médard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," in *Proceedings of the 41st Allerton Conference on Communications, Control and Computing*, 2003.

[20] D. C. Kozen, *Theory of Computation*. Springer-Verlag London Ltd., 2006.