Design of Communication Networks for Distributed Computation with Privacy Guarantees

Sérgio Pequito [⋆,†] Soummya Kar [⋆] Shreyas Sundaram [§] A. Pedro Aguiar [†,‡]

Abstract— **In this paper we address a communication network design problem for distributed computation with privacy guarantees. More precisely, given a possible communication graph between different agents in a network, the objective is to design a protocol, by proper selection of the weights in the dynamics induced by the communication graph, such that 1) weighted average consensus of the initial states of all the agents will be reached; and 2) there are privacy guarantees, where each agent is not able to retrieve the initial states of non-neighbor agents, with the exception of a small subset of agents (that will be precisely characterized). In this paper, we assume that the network is cooperative, i.e., each agent is passive in the sense that it executes the protocol correctly and does not provide incorrect information to its neighbors, but may try to retrieve the initial states of non-neighbor agents. Furthermore, we assume that each agent knows the communication protocol.**

I. INTRODUCTION

Distributed computation has gained renewed interest due to the increase of data being exchanged in large-scale spatially distributed systems. Such as networks of sensors, multi-agent networks and the smart grid [1]. Several schemes have been proposed for the distributed calculation of a function at some agent (not necessarily the same function) that depends on the entire state of a system, for instance, using *consensus* [2] or *state retrieval* [3]. Nevertheless, in several real world scenarios, not all agents are willing to reveal their information. Secure multi-party computation is defined as the problem of n agents computing an agreed upon function of their inputs (initial states) in a secure way, where security means guaranteeing the correctness of the output as well as the privacy of the agents' inputs [4]. Two basic models of communication have been considered in the literature. In the *cryptographic model* [5], all players are assumed to have access to messages exchanged between players, and hence security can only be guaranteed in a cryptographic sense, i.e. assuming that the adversary cannot solve some computational problem. In the *information-theoretic model* [6], [7], it is assumed that the players can communicate over pair-wise secure channels, and security can then be guaranteed even when the adversary has unbounded computing power. The implementation of these protocols can be complex, although some successful applications are known, for instance [8]. In [9] the notion of competitive privacy in multi-agent state estimation was studied from an information-theoretic viewpoint. The setup considered in [9] involved a multi-agent network in which the agents obtain noisy observations of linear combinations of their local and neighboring states and the goal was to design a network communication-computation scheme such that each agent recovers its local state with high fidelity without learning much about the states of other agents.

In this paper, motivated by the simplicity of certain algorithms proposed for distributed computation (such as consensus and state retrieval), we ask the following questions: *Is it possible to derive weighted average consensus protocols ensuring* privacy *of the initial state shared by the agents in a network? If so, which* privacy guarantees *can be obtained?*

Hereafter, we assume that communications are synchronous and bilateral, i.e., if agent i transmits to agent j then agent j also transmits to agent i. This is a common scenario when information is broadcasted in a network of sensors or in social networks, to name a few. Under this setup, the communication graph (or topology) can be understood as an undirected graph, or equivalently, a bidirectional directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ contains the vertices representing the agents and $\mathcal{E}$ the pair of vertices of agents that can communicate with each other. The induced communication protocol consists of an *averaging rule* given by

$$x^i_{k+1} = \sum_{j \in \mathcal{N}_i} w_{i,j} x^j_k, \tag{1}$$

where $x^i_k \in \mathbb{R}$ denotes the state of agent i at time k, the neighbors' indices are given by $\mathcal{N}_i = \{j : (j,i) \in \mathcal{E}\}$, and $w_{i,j} \in \mathbb{R}$ weighs the state of agent j received by agent i.

Without considering any privacy guarantees, given the dynamics presented in (1) constrained to the communication graph (i.e., $w_{i,j} = 0$ if $(j,i) \notin \mathcal{E}$), it is well known how to select the weights such that consensus is obtained, see for instance [1]. Next, we will enhance the consensus protocols with some privacy guarantees. To achieve this goal, we make the following assumptions: (i) we assume that communication channels are secure and cannot be compromised. Thus, every agent can only access the information that he receives; (ii) the network is cooperative, i.e., each agent is *passive* in the sense that he knows the entire communication protocol and executes it correctly (i.e., he does not provide incorrect information to its neighbors); (iii) any agent in the network may try to infer the initial states of the agents that are not its neighbors (via the information received from its own neighbors over time).

Now, notice that (1) can be rewritten as

Support for this research was provided by the Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) through the Carnegie Mellon Portugal Program under Grant SFRH/BD/33779/2009, and by projects CONAV/FCT-PT (PTDC/EEACRO/113820/2009), FCT (PEst-OE/EEI/LA0009/2011), MORPH (EU FP7 No. 288704) and NSF grant 1306128.

[⋆] Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213

[†] Department of Electrical and Computer Engineering, Institute for System and Robotics, Instituto Superior Técnico, Technical University of Lisbon, Lisbon, Portugal

[§] Department of Electrical and Computer Engineering, University of Waterloo, Canada

[‡] Department of Electrical and Computer Engineering, Faculty of Engineering, University of Porto, Porto, Portugal

$$x_{k+1} = W x_k \tag{2}$$

where $x \in \mathbb{R}^n$ is the collection of the scalar states of the n agents in the network. Furthermore, the averaging rule (1) is performed at each agent and each agent knows the communication protocol (by assumption), which implies that each agent i knows each individual term in (1). Hence, each agent i has access to the following measurements

$$y_k^i = I_n^i x_k, \quad i = 1, \ldots, n \tag{3}$$

where I_n^i corresponds to the rows of the $n \times n$ identity matrix with indices in $\bar{\mathcal{N}}_i = \mathcal{N}_i \cup \{i\}$. In other words, an agent i has access to its own state and the state of agent j, if agent j belongs to its neighborhood.

Thus, given the linear system (2)-(3) an agent i can retrieve an arbitrary initial state of the system if (2)-(3) is *observable* [3]. Alternatively, if the system is not observable, it is possible to retrieve some initial states. Those states belong to the so-called *observable subspace*, that depends on W and I_n^i, which we denote by $\mathcal{O}(W, I_n^i)$.

Ergo, given a communication graph $\mathcal{G}$ (ideally) we would like to determine a *communication protocol*, i.e., the weights in W (under the constraint that $w_{i,j} = 0$ if $(j,i) \notin \mathcal{E}$) such that each agent i, with $i \in \{1, \ldots, n\}$, achieves *weighted average consensus* on the initial conditions of all agents and

$$e_j \notin \mathcal{O}(W, I_n^i) \text{ for all } j \notin \mathcal{N}_i, \tag{4}$$

where e_j denotes the jth canonical vector in $\mathbb{R}^n$. In other words, agent i cannot retrieve the initial state of agent j if agent j is not a neighbor of agent i. Hereafter, we consider a related problem, i.e., we analyze the impact that the design of W has on the observability subspace $\mathcal{O}(W, I_n^i)$.

Problem Statement

We propose to design W such that $x_k \xrightarrow{k\to\infty} \mathbf{1} v_l^T x_0$, with left-eigenvector v_l normalized to satisfy $\mathbf{1}^\top v_l = 1$, and

$$\text{rank } \mathcal{O}(W, I_n^i) \ll n, \text{ for } i = 1, \cdots, n, \tag{5}$$

i.e., the dimension of the subspace comprising all the possible initial states that can be inferred, see for instance [10]. Since the number of canonical basis vectors contained in the row space of $\mathcal{O}(W, I_n^i)$ is upper bounded by its rank, minimizing the rank of the matrix provides a means to limit the number of non-neighbor initial states that are recoverable. ◇

To address the above problem, we use structural systems theory [11]. Such tools have not been used before to address privacy issues. Although, protocols with privacy guarantees have been previously addressed, for instance in [12], [13], [14], these commonly consider the injection of random offsets into the agents' state. More precisely, in [13] the offsets are sampled from a zero-mean distribution and introduced once, those will cancel out in average if the number of agents is large enough, although exact computation of the weighted average may not be possible. In addition, since the introduction of an offset is persistent and its model known to all agents, additional attention has to be paid to ensure that stochastic estimation tools cannot be used to retrieve the initial state of the agents [15]. Alternatively, in [12] a similar approach is considered, using ideas borrowed from differential privacy, and a trade-off between privacy and accuracy in probabilistic terms is derived. In [14] the offset is introduced a finite number of times and it is chosen by each agent under the constraint that it has to sum up to a value previously decided by a third party. Further, these values should sum up to zero which implies they will average out as time evolves. Similarly, additional care is needed to ensure that the system is not strongly observable, in other words, it is not observable under unknown inputs [16]. Moreover, [14] provides a verification method to classify which initial states cannot be recovered if an agent is considered to be malicious-curious (i.e., it does not change the protocol or injects false data into the network). The present paper considers the design of the network weights (dynamics) such that all agents are potentially malicious-curious and does not required to inject noise in its state to disguise its initial state. Finally, we note other approaches have been proposed for the case where the agents are active (or malicious), i.e., do not follow the communication protocol [17].

The main contribution of this paper is to design communication protocols that achieve weighted average consensus on the initial states of the agents, while allowing an agent to retrieve only a small subset of the initial states of the agents that are not its neighbors.

The rest of the this paper is organized as follows. Section II provides some preliminary concepts and results. In Section III we present intermediate results that study the interplay between the structure of a graph and the achievable rank of the observable subspace for each agent. Subsequently, in Section IV we present the main technical results (with proofs can be found in [18]), followed by an illustrative example in Section V. Conclusions and discussion avenues for further research are presented in Section VI.

II. PRELIMINARIES AND TERMINOLOGY

In this section, we review some concepts of structural systems [19]. Consider a linear time invariant system (LTI) described as

$$x_{k+1} = W x_k, \quad y_k = C x_k \tag{6}$$

where $x_k \in \mathbb{R}^n$ represents the state and $y_k \in \mathbb{R}^m$ denotes the measured output. If the system (6) is observable, then we say that the pair (W, C) is observable. A system of the form (6) is said to be a *structural system* if each entry of W and C is viewed either a fixed zero or an arbitrary (independent) free parameter. Further, if almost all numerical realizations with the same structured system are observable then we say that (6) is *structurally observable* [11].

Structural systems provide an efficient representation of the system as a directed graph (digraph). Each digraph is associated with a set of *vertices* $\mathcal{V}$ and a set of *directed edges* $\mathcal{E}$ of the form (v_i, v_j) where $v_i, v_j \in \mathcal{V}$. We represent the state digraph by $\mathcal{D}(W) = (\mathcal{X}, \mathcal{E}_{\mathcal{X},\mathcal{X}})$, i.e., the digraph that comprises only the state variables as vertices denoted by $\mathcal{X} = \{x^1, \cdots, x^n\}$ and a set of directed edges between the state vertices denoted by $\mathcal{E}_{\mathcal{X},\mathcal{X}} = \{(x^i, x^j) : x^i, x^j \in \mathcal{X} \text{ and } w_{j,i} \neq 0\}$, where $w_{i,j}$ denotes the entry in the ith row and jth column in matrix W. Similarly, we represent the system digraph by $\mathcal{D}(W, C) = (\mathcal{X} \cup \mathcal{Y}, \mathcal{E}_{\mathcal{X},\mathcal{X}} \cup \mathcal{E}_{\mathcal{X},\mathcal{Y}})$, where $\mathcal{Y} = \{y^1, \cdots, y^m\}$ and $\mathcal{E}_{\mathcal{X},\mathcal{Y}} = \{(x^i, y^j) : y^j \in \mathcal{Y},\, x^i \in \mathcal{X} \text{ and } c_{j,i} \neq 0\}$, where $C = [c_{j,i}]$.

Given a digraph $\mathcal{D}$, a digraph $\mathcal{D}_s = (\mathcal{V}_s, \mathcal{E}_s)$ such that $\mathcal{V}_s \subset \mathcal{V}$ and $\mathcal{E}_s \subset \mathcal{E}$ is said to be *subgraph* of $\mathcal{D}$. If $\mathcal{V}_s = \mathcal{V}$, $\mathcal{D}_s$ is said to *span* $\mathcal{D}$. A sequence of edges $\{(v_1, v_2), (v_2, v_3), \ldots, (v_{k-1}, v_k)\}$ is a *directed path*. If all the vertices in a directed path are distinct then it is said to be an *elementary path*. A directed path with $v_k = v_1$ ($k \geq 1$) and all other vertices distinct is called a *cycle*. A digraph $\mathcal{D}$ is said to be strongly connected if there exists a directed path between any two vertices. We also use operations over graphs, such as difference, union and intersection: given two graphs $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1), \mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$, the difference is given by $\mathcal{G}_1 \setminus \mathcal{G}_2 = (\mathcal{V}_1 \setminus \mathcal{V}_2, \mathcal{E}_1 \setminus \mathcal{E}_2)$, the intersection by $\mathcal{G}_1 \cap \mathcal{G}_2 = (\mathcal{V}_1 \cap \mathcal{V}_2, \mathcal{E}_1 \cap \mathcal{E}_2)$ and union by $\mathcal{G}_1 \cup \mathcal{G}_2 = (\mathcal{V}_1 \cup \mathcal{V}_2, \mathcal{E}_1 \cup \mathcal{E}_2)$. For any two vertex sets $\mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{V}$, we define the *bipartite graph* $\mathcal{B}(\mathcal{S}_1, \mathcal{S}_2, \mathcal{E}_{\mathcal{S}_1,\mathcal{S}_2})$ associated with $\mathcal{D} = (\mathcal{V}, \mathcal{E})$, to be a directed graph (bipartite), whose vertex set is given by $\mathcal{S}_1 \cup \mathcal{S}_2$ and the edge set $\mathcal{E}_{\mathcal{S}_1,\mathcal{S}_2}$ by $\mathcal{E}_{\mathcal{S}_1,\mathcal{S}_2} = \{(s_1, s_2) \in \mathcal{E} \ : \ s_1 \in \mathcal{S}_1, s_2 \in \mathcal{S}_2 \ \}$.

Given $\mathcal{B}(\mathcal{S}_1, \mathcal{S}_2, \mathcal{E}_{\mathcal{S}_1,\mathcal{S}_2})$, a matching M corresponds to a subset of edges in $\mathcal{E}_{\mathcal{S}_1,\mathcal{S}_2}$ that do not share vertices, i.e., given edges $e = (s_1, s_2)$ and $e' = (s_1', s_2')$ with $s_1, s_1' \in \mathcal{S}_1$ and $s_2, s_2' \in \mathcal{S}_2$, $e, e' \in M$ only if $s_1 \neq s_1'$ and $s_2 \neq s_2'$. A maximum matching M^* may then be defined as a matching M that has the largest number of edges among all possible matchings. The vertices in $\mathcal{S}_1$ and $\mathcal{S}_2$ are *matched vertices* if they belong to an edge in the maximum matching M^*, otherwise, we designate the vertices as *unmatched vertices*. If there are no unmatched vertices, we say that we have a *perfect matching*. It is to be noted that a maximum matching M^* may not be unique.

For ease of referencing, in the sequel, the term *left-unmatched vertices* (w.r.t. $\mathcal{B}(\mathcal{S}_1, \mathcal{S}_2, \mathcal{E}_{\mathcal{S}_1,\mathcal{S}_2})$ and a maximum matching M^*) will refer to only those vertices in $\mathcal{S}_1$ that do not belong to a matched edge in M^*.

Lemma 1 (Minimal Path and Cycle Decomposition [19]): Consider the digraph $\mathcal{D}(W) = (\mathcal{X}, \mathcal{E}_{\mathcal{X},\mathcal{X}})$ and let M^* be a maximum matching associated with the bipartite graph $\mathcal{B}(\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X},\mathcal{X}})$. Then, the digraph $\mathcal{D}^* = (\mathcal{X}, M^*)$ constitutes a disjoint union of cycles and elementary paths comprising only state variables (by convention an isolated state vertex is an elementary path), with the ends in the left-unmatched vertices of M^*, that span $\mathcal{D}(W)$. Moreover, such a decomposition is *minimal*, in the sense that no other spanning subgraph decomposition of $\mathcal{D}(W)$ into elementary paths and cycles contains a strictly smaller number of elementary paths. ◇

With some abuse of terminology, the notion of an elementary path extends to undirected graphs, where the edges are considered to be undirected. Given an undirected connected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, the degree of a vertex consists of the number of neighbors a vertex shares an undirected edge with. In addition, in this paper, when we use the term undirected graphs we assume them to be loopless, i.e., without self-loops. If there is exactly one (undirected) elementary path between any pair of vertices $u, v \in \mathcal{V}$, then $\mathcal{G}$ is *a tree*, in other words a tree is a connected graph with the minimum number of edges. In a tree, any vertex that has only one neighbor is referred to as *a leaf*. A special case of a tree is given next.

Definition 1 (Star): A star is a tree with at least two vertices where all vertices but one have degree one. In addition, all vertices with degree one are referred to as *corners*, whereas the remaining vertex is referred to as the center. ◇

Remark 1: In the remaining of the paper we will identify stars as subgraphs of an original graph, and eventually a tree. Therefore, we have chosen to introduce the notion of corners instead of the standard notion of leaves since the corners of stars may not necessarily correspond to the leaves of a tree when subgraphs are considered. ◇

Definition 2 ([20]): (Set Covering Problem) Given a set of m elements $\mathcal{U} = \{1, 2, \ldots, m\}$ and a set of n sets $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$ such that $\mathcal{S}_i \subset \mathcal{U}$, with $i \in \mathcal{J} = \{1, \cdots, n\}$, and $\displaystyle\bigcup_{i \in \mathcal{J}} \mathcal{S}_i = \mathcal{U}$. The set covering problem consists of finding the smallest set of indices $\mathcal{I}^* \subseteq \mathcal{J}$ such that $\mathcal{U} \subseteq \displaystyle\bigcup_{i \in \mathcal{I}^*} \mathcal{S}_i$. ◇

The set covering problem is used in the present paper to find the structure of the graph used in the communication protocol.

Weighted Consensus Conditions

Recall that a matrix is called nonnegative if all its entries are greater than or equal to zero. Additionally, a matrix is called primitive if there exists a natural number k such that W^k has all entries greater than zero. Thus, from the Perron-Frobenius theorem we have the following result.

Theorem 1 ([21], [1]): Let W be a nonnegative row-stochastic primitive matrix with left eigenvector v_l normalized to satisfy $\mathbf{1}^\top v_l = 1$. Then, v_l is strictly positive (i.e., all components of v_l are positive) and we have $\lim_{k \to \infty} W^k = v_r v_l^T$. ◇

Generic Dimensions of Observable Subspaces

We now recover some of the results in the context of generic observable subspaces [22]. The *generic dimension of observable subspaces* (GDO) is the maximum dimension that the observable subspaces can achieve over all numeric realizations of the system metrics that respect the sparsity structure.

Lemma 2 (Generic Dim. Observability Space [22]): Let $\mathcal{D}(W)$ be a strongly connected graph. Given a non-zero output matrix C, the generic dimension of the observable space (GDO), denoted by $gdo(\mathcal{D}(W, C))$ satisfies $gdo(\mathcal{D}(W, C)) = |M_o^*|$, where M_o^* is a maximum matching of the bipartite graph associated with $\mathcal{D}(W, C)$. □

Notice that the GDO is defined for a specific graph structure and a collection of measurements. Hereafter, with some abuse of terminology, when referring to the GDO we implicitly consider that the output matrix C is of the form (3), which is closely related with the structure of the communication graph. More precisely, given a communication graph, then the structure of the dynamics and the output matrices is fixed, where the latter consists of dedicated outputs assigned to the agent i and its neighbors.

III. Intermediate Results

In this section we study the interplay between the graph structure and the generic dimension of the observability space (GDO), given a specified set of dedicated outputs or measurements. Towards this goal, we proceed as follows: (1) we introduce the notion of *generic dimension of the state space* (GDS) induced by the system digraph, that is closely related to the notion of the GDO; (2) we explicitly characterize these quantities for specific classes of system digraphs, such as trees, upon which we provide a decomposition that we shall refer to as *constellation*.

We start by introducing the notion of generic dimension of the state space (GDS).

Definition 3 (Generic Dim. State Space (GDS)): Given a strongly connected graph $\mathcal{D}(W)$, the generic dimension of the state space is given by $gds(\mathcal{D}(W)) = |M^*|$, where M^* corresponds to a maximum matching of the bipartite graph associated with $\mathcal{D}(W)$ (i.e., the state bipartite graph). $\diamond$

Now, consider the following result (see [19] and [23]) which relates the GDS of a system digraph $\mathcal{D}(W)$ with the GDO associated with an output placement configuration C.

Proposition 1 ([23], [19]): Given state and output matrices W and C, let $\mathcal{D}(W, C)$ be the system digraph. Suppose C is non-zero and comprises only dedicated outputs, and the digraph $\mathcal{D}(W)$ is strongly connected. Then, the associated GDS and GDO are related as follows: $gdo(\mathcal{D}(W, C)) = gds(\mathcal{D}(W)) + l$, where l corresponds to the maximum number of left-unmatched vertices (state variables), with respect to all possible maximum matchings of the state bipartite graph $\mathcal{B}(W)$, measured by the dedicated outputs. $\diamond$

With some abuse of notation, we use $gdo(\mathcal{G}, C) = gdo(\mathcal{D}(W, C))$, where $\mathcal{D}(W)$ is the state digraph induced by $\mathcal{G}$.

Next, we introduce the notion of *loopable* vertices. Broadly speaking, loopable vertices (formally defined below) are vertices to which we can add a self-loop without increasing the GDS of the state digraph. Recall that our objective is to design W that solves (4)-(5), and hence, we simultaneously need to ensure that the associated GDS (consequently, GDO, by Proposition 1) is small (such that privacy requirements are met) and that W is a primitive matrix (to guarantee weighted consensus). It is well known that primitivity can be ensured by adding self-loops in W, however, in general, such self-loops may increase the GDS. This motivates our study of loopable vertices, as introduced in the following.

Definition 4 (Loopable-vertex): Consider a state digraph $\mathcal{D}(W) = (\mathcal{X}, \mathcal{E}_{\mathcal{X},\mathcal{X}})$ and its associated bipartite graph with a maximum matching of size p. We say that a vertex $x \in \mathcal{X}$ is a loopable-vertex if the maximum matchings of the bipartite graph associated with $\mathcal{G}^o = (\mathcal{X}, \mathcal{E}_{\mathcal{X},\mathcal{X}} \cup \{(x, x)\})$ have size p. $\diamond$

Remark 2: Such loopable vertices are important in terms of ensuring primitivity of the W matrix that we design, and, additionally, provide more degrees of freedom in selecting the numerical entries of W, for instance, with a view to obtaining faster convergence. $\diamond$

The notion of a loopable vertex is complemented with the notion of *critical* vertices, i.e., those vertices in which a self-loop or the assignment of a dedicated output increases the maximum matching associated with the state and system bipartite graphs respectively. More precisely, we have the following definition.

Definition 5 (Critical Vertex): Consider a state digraph $\mathcal{D}(W) = (\mathcal{X}, \mathcal{E}_{\mathcal{X},\mathcal{X}})$ and let $\mathcal{B} = (\mathcal{X}, \mathcal{X}, \mathcal{E}_{\mathcal{X},\mathcal{X}})$ be the associated bipartite graph. A subset of state variables $\mathcal{X}' \subset \mathcal{X}$ is called a critical set if there exists a maximum matching M of $\mathcal{B}$ such that $\mathcal{X}' \subset \mathcal{U}_L$, where $\mathcal{U}_L$ denotes the set of left-unmatched vertices of M. Moreover, a *maximal critical subset* is a subset of critical vertices such that no other subset of critical vertices has more elements. $\diamond$

Remark 3: The notions of loopable and critical vertices are complementary in the sense that vertices that are not critical are loopable. Those notions are, to some extent, related with the study presented in [23], where the loopable vertices would correspond to the state variables that always belong to the left endpoint of an edge in a maximum matching, for all possible matchings. On the other hand, in [23] only single state variables that may not belong to the edges in some maximum matching were considered, whereas the maximal critical subset consists of those variables with respect to some maximum matching. $\diamond$

We now study the interplay between structure and its associated GDS when a tree is considered. In particular, it is useful to consider the following decomposition of the tree.

Definition 6 (k-constellation): Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a tree graph and, $\mathcal{L} = \{l_1, \ldots, l_p\}$ its p leaves. Additionally, consider all the edges with one end-point in the leaves, i.e., $\mathcal{P} = \{(v, l) \in \mathcal{E} : \ v \in \mathcal{V}, \ l \in \mathcal{L}\}$. Further, let $\mathcal{V}_P = \{v : (v, l) \in \mathcal{P}, l \in \mathcal{L}\} = \{v_1, \ldots, v_k\}$ correspond to the left end-points of the edges in $\mathcal{P}$. Then each subgraph $\mathcal{S}_i = (\mathcal{V}_i, \mathcal{E}_i)$, where $i = 1, \ldots, k$, with $\mathcal{V}_i = \{v_i\} \cup \{l : (v_i, l) \in \mathcal{P}, \ l \in \mathcal{L}\}$ for each $v_i \in \mathcal{V}_P$ and $\mathcal{E}_i = \{(v_i, l) \in \mathcal{P} : l \in \mathcal{V}_i\}$, is a *star* and each v_i is a center. Moreover, consider each strongly connected component of $\mathcal{G} \backslash \{\bigcup_{i=1,\ldots,k} \mathcal{S}_i\}$, referred to as a *chord*. Denote by $\mathcal{C}_j$, $j = 1, \ldots, m$, the set of chords. The collection $\{\mathcal{S}_i\}$ and $\{\mathcal{C}_j\}$ constitutes a unique decomposition of the tree graph $\mathcal{G}$ into a disjoint union of stars and chords, referred to as a *constellation*. $\diamond$

A constellation, arising from the decomposition of a tree graph as described in Definition 6, is said to be a k-constellation if it is comprised of k (disjoint) stars.

Notice that, by Definition 1, we may have degenerate stars with two vertices, in which case any vertex can play the role of a center or corner (but not both simultaneously), and the neighbors of a center in a star subgraph of the constellation is a subset (possibly strict) of the corresponding set of neighbors in the original graph.

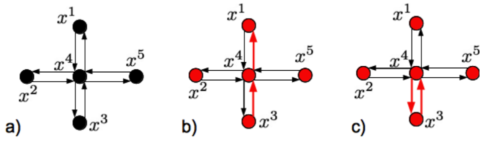

Fig. 1. In a) we depict a star with five vertices, and in b) and c) we have two possible decompositions using Lemma 1, both with two edges in a maximum matching of the state bipartite graph and hence with GDS (see Definition 3) equal to 2.

We can further classify chords as follows.

Definition 7 (Even/Odd-Chords): A chord is said to be even if the maximum matching associated with its bipartite graph is a perfect matching. Otherwise, it is said to be an odd-chord. $\diamond$

We now show that stars have the minimum GDS among all possible connected bidirectional graphs.

Lemma 3: Let $[\mathcal{G}_n]$ be the class of all connected bidirectional digraphs (without self-loops) with $n \geq 2$ vertices. Then, the star $\mathcal{S}_n$ on n vertices has the smallest GDS in $[\mathcal{G}_n]$, i.e., $gds(\mathcal{S}_n) \leq gds(\mathcal{G})$ for all $\mathcal{G} \in [\mathcal{G}_n]$. $\diamond$

We now state the relationship between the GDS of a star state graph and the GDO associated with an agent i, $i = 1, \cdots, n$, which receives dedicated output measurements of its own state and those of its neighbors.

Lemma 4: Let $\mathcal{S}_n = (\{x_1, \cdots, x_n\}, \mathcal{E})$ be a star with $n \geq 2$ vertices where x_1 is the center and the remaining vertices are corners. Then

(i) $gdo_i(\mathcal{S}_n) = 3$ if $i = 2, \cdots, n$ and $n \geq 3$;

(ii) $gdo_i(\mathcal{S}_n) = n$, otherwise;

where $gdo_i(\mathcal{S}_n) = gdo(\mathcal{S}_n, I_n^i)$. $\diamond$

So far we have considered the relationship between the GDS and GDO of a single star. Although stars minimize the GDS (and hence, are desirable from the privacy viewpoint), it is hardly the case that an inter-agent communication graph is spanned by a single star. However, a connected inter-agent communication graph is always spanned by a constellation, which motivates us to extend the previous development to structures consisting of combinations or unions of stars, and, in particular, the case of two stars connected to each other by an edge.

Lemma 5: Let $\mathcal{S}_{n_1} = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{S}_{n_2} = (\mathcal{V}_2, \mathcal{E}_2)$ be two disjoint stars with $n_1, n_2 > 2$ vertices respectively. If $\mathcal{G} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \{e\}$ where $e = (v_1, v_2)$ with $v_1 \in \mathcal{V}_1$ and $v_2 \in \mathcal{V}_2$, then two cases are possible:

(i) if v_1, v_2 are corners in $\mathcal{S}_{n_1}$ and $\mathcal{S}_{n_2}$ respectively, then $gds(\mathcal{G}) = gds(\mathcal{S}_1) + gds(\mathcal{S}_2) + 2 = 6$;

(ii) $gds(\mathcal{G}) = gds(\mathcal{S}_1) + gds(\mathcal{S}_2) = 4$, otherwise. $\diamond$

Remark 4: From Lemma 5 we conclude that given two disjoint stars, if we want to add an edge to connect them to obtain a connected graph and attain minimum GDS, we should connect them from one of the centers to the corner/center of the other. Conversely, if two disjoint stars are connected through their corners, the final connected graph has an even chord and consequently the GDS increases. If the two stars are connected through their centers, no chord is created and we have a 2-constellation. Finally, if a center is connected to a corner, then an odd chord consisting of a single vertex is created in the constellation thus obtained. $\diamond$

Further, we notice in order to obtain a constellation given a disjoint collection of stars, we can proceed inductively as described in Remark 4. This motivates our algorithmic procedure in Algorithm 1, which, given an arbitrary connected graph, aims to construct a spanning constellation subgraph culminating in the privacy guarantees described in the main result (Theorem 3) of this paper.

IV. Main Results

In the previous section we obtained the relationship between GDS and GDO for stars. Furthermore, by iteratively considering a connection of stars, we obtained a constellation. Hereafter we start by showing the relation between the GDS and GDO in the case of an arbitrary constellation (Theorem 2). Then, among the possible constellations that span the original communication digraph we conclude that we want to find the constellation that has the smaller number of stars and chords. In fact, when possible, stars are preferred to chords since stars minimize the GDS (see Lemma 3). Further, we provide an upper-bound for the GDO of the constellation when a specific agent is considered, see Corollary 1. Upon this upper-bound we provide an algorithm (Algorithm 1) that further minimizes the upper-bound, see Theorem 3. Finally, once the structure of the loopless graph $\mathcal{G}'$ that ensures a small GDO for the different agents is determined, we show that exists a real matrix W that ensures weighted average consensus. To this goal, we consider the matrix structure W' induced by G' with its loopable vertices, to which there exists a numeric realization with the same structure ensuring our goal, see Theorem 4.

We start by noticing that for a constellation the GDO and GDS are related as follows.

Theorem 2: Let $\mathcal{D}(W')$ be a k-constellation that spans the original communication graph $\mathcal{G}$, with stars $\{\mathcal{S}_l\}_{l=1,\ldots,k}$ and chords $\{\mathcal{C}_j\}_{j=1,\ldots,m}$. The GDO associated with $\mathcal{D}(W', I_n^i)$, i.e., for an agent i, is given as

$$gdo_i(\mathcal{D}(W')) = \sum_{l=1}^{k} \left(gds(\mathcal{S}_l)\ + s_l^i\right) + \sum_{j=1}^{m} \left(gds(\mathcal{C}_j) + c_j^i\right),$$

where $gdo_i(\mathcal{D}(W')) = gdo(\mathcal{D}(W', I_n^i))$, and s_l^i and c_j^i denote the largest number of critical vertices in a critical subset of the star $\mathcal{S}_l$ and chord $\mathcal{C}_j$ respectively, that an agent i measures. $\diamond$

In particular, we obtain an upper-bound to $gdo_i(W')$ as presented next.

Corollary 1: Let $\mathcal{D}(W')$ be a k-constellation that spans the original communication graph $\mathcal{G}$ with stars $\{\mathcal{S}_l\}_{l=1,\ldots,k}$ and chords $\{\mathcal{C}_j\}_{j=1,\ldots,m}$. The GDO associated with $\mathcal{D}(W', I_n^i)$, i.e., for an agent i, is given as follows

$$gdo_i(W') \leq \sum_{l=1}^{k} gds(\mathcal{S}_l) + \sum_{j=1}^{m} gds(\mathcal{C}_j) + |\mathcal{N}_i|,$$

where $gdo_i(\mathcal{D}(W')) = gdo(\mathcal{D}(W', I_n^i))$. $\diamond$

Remark 5: From Theorem 2 and Corollary 1, we have that, in order to minimize the GDO for an agent, we need to minimize the cumulative GDS of stars and chords in a spanning constellation subgraph of the original communication graph $\mathcal{G}$. Since stars minimize the GDS among all bidirectional connected graphs (Lemma 3), from the design perspective, we prefer spanning constellations of $\mathcal{G}$ with more stars and fewer chords. Additionally, for stars with more than 2 vertices the GDS is invariant to the number of vertices (Lemma 4), and, hence, we would like to determine spanning constellations having the smallest number of stars with more than two vertices. $\diamond$

Motivated by Remark 5, we propose Algorithm 1 that provides a procedure aimed at obtaining the spanning constellation subgraph of $\mathcal{G}$ that minimizes the cumulative sum GDS of stars and chords (and hence the upper bound in (7)) among all possible constellation subgraphs of $\mathcal{G}$. Subsequently, we can upper bound the worst case GDO associated with the constellation generated by Algorithm 1 as follows.

Theorem 3: Given an initial bidirectional communication graph $\mathcal{G}$, let $\{\mathcal{S}_j^*\}_{j\in\mathcal{J}^*}$ be the collection of spanning stars obtained in Algorithm 1. Then, denoting by $\mathcal{D}''$ the output constellation of Algorithm 1, we have

$$gdo_i(\mathcal{D}'') \leq \sum_{j \in \mathcal{J}^*} gds(\mathcal{S}_j^*) + \bar{e} + |\mathcal{N}_i| \leq 4|\mathcal{J}^*| + |\mathcal{N}_i| \quad (7)$$

where $gds(\mathcal{S}) = 0$ if $\mathcal{S}$ is a degenerated star with a single vertex, $gdo_i(\mathcal{D}'') = gdo(\mathcal{D}'', I_n^i)$, $\mathcal{J}^*$ denotes the number of stars in $\mathcal{D}''$ (or the number of sets in the minimal set covering in Algorithm 1), and $\bar{e}$ is the cardinality of the subset of edges introduced in the last operation in Algorithm 1 (to ensure connectedness of $\mathcal{D}''$) whose both end-points are corner vertices belonging to distinct stars (i.e., in particular, $\bar{e} = 0$ if $\mathcal{D}'$ in Algorithm 1 is connected.) $\diamond$

Now, note that by the definition of generic rank, any numerical weight matrix W with the structure of $\mathcal{D}''$ satisfies rank $\mathcal{O}(W; I_n^i) \leq gdo_i(\mathcal{D}'')$ for all i. Hence, by Theorem 3, privacy guarantees as far as the distributed computation problem (4)-(5) is concerned, can be ensured by generating an arbitrary W with sparsity $\mathcal{D}''$. However, since we are interested in obtaining weighted consensus, the W needs to satisfy certain additional conditions, specifically primitivity. Note that any W with the structure of D'' is irreducible and hence, to ensure primitivity, it is sufficient to introduce self-weights (or topologically, self-loops) in W. To this end, we study which vertices in D'' are loopable, such that introducing self-loops at these vertices will not increase the GDO of D'' but will ensure that primitive numerical instances W of D'' exist.

Lemma 6: Given a k-constellation, we have the following:

(i) the centers of the k stars are loopable;
(ii) all vertices in an even-chord are loopable;
(iii) all non-critical vertices in odd-chords are loopable. $\diamond$

In particular, note that the set of loopable vertices is non-empty. It is also important to state the dual result, i.e., which self-loops increase the GDS.

Lemma 7: Given a star $\mathcal{S}_n = (\mathcal{V}, \mathcal{E})$ with $n \geq 3$ vertices, adding a self-loop to a corner $v \in \mathcal{V}$, leads to $gds(\mathcal{S}_n^o) = gds(\mathcal{S}_n) + 1$, where $\mathcal{S}_n^o = (\mathcal{V}, \mathcal{E} \cup \{v, v\})$. $\diamond$

To summarize, we have obtained a constellation (given by Algorithm 1) with very small GDS within the possible spanning constellations of the communication graph. In addition, given the constellation, we can identify the loopable vertices (see Lemma 8). Combining the above, it can be readily seen that a weight matrix W, in which the non-zero entries correspond to edges in the constellation $\mathcal{D}''$ (obtained in Algorithm 1) and additional self-weights (non-zero diagonal entries) on the loopable vertices (determined in Lemma 8), achieves agent-wise privacy guarantees as in Theorem 8 while being irreducible with self-loops. Now, primitivity and row-stochasticity can be ensured by selecting the non-zero entries in W such that they are positive and each row in W sums to 1. The above observations lead to the main result of this paper on the design of W in (4)-(5) that achieves weighted consensus with privacy guarantees stated as follows.

Theorem 4: Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a communication graph. Let the subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ of $\mathcal{G}$ be determined using Algorithm 1 and $\mathcal{E}^o$ comprise the edges associated with the self-loops of the loopable vertices. Then there exists a row-stochastic matrix W with $\mathcal{D}(W) \equiv \mathcal{G}'' = (\mathcal{V}, \mathcal{E}' \cup \mathcal{E}^o)$ ensuring weighted average consensus of (2). $\diamond$

Remark 6: For instance, under the hypotheses of Theorem 4, a primitive and row-stochastic W can be designed by simply setting $W_{ij} = 0$ if $(i, j) \notin \mathcal{E}' \cup \mathcal{E}^0$ and choosing the non-zero entries in each row, say the i-th row, to be equal to $1/N_i$ where N_i denotes the number of bidirectional edges including self-loops incident on agent i in the graph $\mathcal{G}''$. Finally, note that under the conditions of Theorem 4, the agents reach weighted consensus on the linear combination $\mathbf{v}_l^\top \mathbf{x}_0$ of the initial states $\mathbf{x}_0$; however, since the left-eigenvector $\mathbf{v}_l$ consists of strictly positive components, any other weighted combination can be achieved by simply rescaling the local initial value at each node appropriately. $\diamond$

ALGORITHM 1: Determine a (connected) constellation spanning the communication graph $\mathcal{G}$

```
Input: Bidirectional communication digraph $\mathcal{D} = (\mathcal{V}, \mathcal{E})$
Output: A constellation (connected bidirectional spanning
        subgraph $\mathcal{D}'$ of $\mathcal{D}$, see Definition 6)

Set $\mathcal{S}_i = \bar{\mathcal{N}_i}(\equiv \mathcal{N}_i \cup \{i\})$ and $\mathcal{U} = \bigcup_{i=1,\ldots,n} \mathcal{S}_i$

Find a solution to the minimum set covering problem with the
sets $\mathcal{S}_i,\ i \in \{1, \ldots, n\}$ and $\mathcal{U}$, which we denote by $\mathcal{I}^*$

Let $\mathcal{S}_i^* = (\mathcal{V}_{\mathcal{S}_i}, \mathcal{E}_{\mathcal{S}_i})$ (with $i \in \mathcal{I}^*$) be a star, consisting of the
vertices with indices in $\mathcal{S}_i$ and edges with one endpoint in $i$.

If the intersection between two stars is non-empty, consider
the following two (exhaustive) scenarios: (i) a center belongs
to the intersection, in which case the connection between the
stars should be established using the edge connecting centers;
else (ii) let $\mathcal{S}_j^c$ and $\mathcal{S}_{j'}^c$ correspond to the sets containing only
the corner vertices of $\mathcal{S}_j^*$ and $\mathcal{S}_{j'}^*$ respectively. If $|\mathcal{S}_i^c \cap \mathcal{S}_j^c|$ is
greater than one, then disregard $|\mathcal{S}_i^c \cap \mathcal{S}_j^c| - 1$ edges ending in
different vertices in $\mathcal{S}_i^c \cap \mathcal{S}_j^c$, and such that the degree of each
star is greater than one if possible (see Remark 5). Denote by
$\mathcal{E}'$ the edges obtained after applying (i)-(ii).

In addition, if $\mathcal{D}' = (\mathcal{V}, \mathcal{E}')$ is disconnected, determine the set
of (bidirectional) edges $\mathcal{E}'' \subset \mathcal{E}$ with minimum size such that
$\mathcal{D}'' = (\mathcal{V}, \mathcal{E}' \cup \mathcal{E}'')$ is strongly connected, where preferentially
using those edges that do not link two corners of the stars
previously found (see Remark 4).
```

In the next section we provide an illustrative example that uses the main results obtained in this section.

V. AN ILLUSTRATIVE EXAMPLE

Consider the communication graph $\mathcal{G}$ depicted in Figure 2, where the directed edges are the communication links between the agents. Further, all vertices have self-loops that are not depicted. To apply Algorithm 1, we create the sets $\mathcal{S}_i \equiv \bar{\mathcal{N}_i} = \{i\} \cup \mathcal{N}_i$, with $i \in \{1, \ldots, 14\}$, and set $\mathcal{U} = \{1, \ldots, 14\}$. A possible solution to the set covering problem is given by the sets $\mathcal{S}_1^*, \mathcal{S}_2^*, \mathcal{S}_3^*$ corresponding to the indices associated with the vertices in each of the dashed gray circles in Figure 3. Now, notice that the sets $\mathcal{S}_l^*$ $(l = 1, 2, 3)$ span $\mathcal{G}$, where $\mathcal{S}_1$ and $\mathcal{S}_3$ intersect at one vertex, but $\mathcal{S}_2$ is disjoint from the rest. Hence, an edge connecting the red star to the others needs to be chosen (from the set of available communication edges given by the initial graph $\mathcal{G}$ to obtain a constellation. Considering Remark 4, in order to minimize the overall GDS, we select the edge connecting the center of the red star to the center of the green star we select, instead of the edges in red that connect corners of different stars. Finally, we obtain the spanning digraph comprising the blue edges, denoted by $\mathcal{E}'$. In addition, notice that the vertex in yellow, although initially was the corner of a star, becomes an odd chord. The obtained constellation thus consists of three stars and a chord. Further, the set $\mathcal{E}''$ contains the edges associated with the self-loops at the loopable vertices, in this case the vertices $1, 2, 3$ corresponding to the centers of each star in the 3-constellation. We finally obtain the digraph $\mathcal{D}' = (\{1, \ldots, 14\}, \mathcal{E}^* \equiv \mathcal{E}' \cup \mathcal{E}'')$ as the output of Algorithm 1. Using Theorem 4 we can now design a W matrix that achieves weighted consensus with agent-wise privacy guarantees as given in (7). The specific W matrix used for this example was obtained by selecting the numerical entries as described in Remark 6.

Furthermore, to show that the privacy guarantees hold and information leakage (see (4)-(5)) is minimized. For instance,

$$\mathcal{R} = [8\ 8\ 8\ 8\ 9\ 8\ 9\ 9\ 9\ 9\ 9\ 9\ 9\ 9]$$

where each entry is given by $\mathcal{R}_j = \text{rank}[\mathcal{O}(W, I_n^4);\ e_j^T]$. Thus, if the entry $\mathcal{R}_j > \mathcal{R}_i$, then the canonical vector e_j associated with the initial data at agent j does not belong to the observability subspace of agent i, i.e., agent i cannot retrieve the initial condition of agent j.

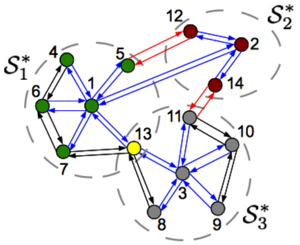

Fig. 2. In this figure the communication digraph $\mathcal{G}$ is represented by the bidirectional edges, whereas the sets of vertices denoted by $\mathcal{S}_i^*$, $i = 1, 2, 3$, correspond to a solution of the minimum set covering problem posed in Algorithm 1. As stated in Algorithm 1, we create stars based on these sets, which are depicted by green/red/gray with the yellow vertex belonging to both $\mathcal{S}_1^*, \mathcal{S}_3^*$. The red star is seen to be disjoint from the others. Hence, an edge connecting the red star to the others needs to be chosen. Considering Remark 4, in order to minimize the overall GDS, we select the edge connecting the center of the red star to the center of the green star instead of the edges in red that connect corners of different stars. In addition, notice that the vertex in yellow, although initially was a corner of a star, becomes a chord, in particular an odd-chord, in the final constellation. The edges in blue depict the edges in set $\mathcal{E}'$ corresponding to the constellation found by Algorithm 1.

VI. CONCLUSIONS AND FURTHER RESEARCH

In this paper we analyzed the interplay between a specific network and its observability subspace. We proposed a communication protocol that achieves weighted average consensus on the initial states of the agents in the network, while allowing an agent to retrieve only a small subset of the initial states of the agents that are not its neighbors. It remains to explore if there exist additional subclasses of network topologies where similar approaches can be used, as well as to understand to the full extent of how the choice of parameters can lead to a smaller subset of initial states being recovered. For this approach it may be worth noticing that Algorithm 1 is related to the dominating set problem, which can be used to obtain new results. Additionally, it would be interesting to understand if specific numerical realizations lead to subspaces where privacy is fully ensured, i.e., each agent can only retrieve the initial state of its neighbors.

REFERENCES

[1] 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.

[2] F. Bullo, J. Cortes, and S. Martinez, *Distributed Control of Robotic Networks*. Princeton University Press, 2009.

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

[4] V. Kolesnikov, "Advances and impact of secure function evaluation," *Bell Labs Technical Journal*, vol. 14, no. 3, pp. 187–192, 2009.

[5] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, ser. STOC '87. New York, NY, USA: ACM, 1987, pp. 218–229.

[6] D. Chaum, C. Crépeau, and I. Damgard, "Multiparty unconditionally secure protocols," in *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, ser. STOC '88. New York, NY, USA: ACM, 1988, pp. 11–19.

[7] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computation," in *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, ser. STOC '88. New York, NY, USA: ACM, 1988, pp. 1–10.

[8] P. Bogetoft, D. Christensen, I. Damgrd, M. Geisler, T. Jakobsen, M. Krigaard, J. Nielsen, J. Nielsen, K. Nielsen, J. Pagter, M. Schwartzbach, and T. Toft, "Secure multiparty computation goes live," in *Financial Cryptography and Data Security*, ser. Lecture Notes in Computer Science, R. Dingledine and P. Golle, Eds. Springer Berlin Heidelberg, 2009, vol. 5628, pp. 325–343.

[9] L. Sankar, S. Kar, R. Tandon, and H. V. Poor, "Competitive privacy in the smart grid: An information-theoretic approach." in *IEEE Second International Conference on Smart Grid Communications*, 2011, pp. 220–225.

[10] J. Hespanha, *Linear Systems Theory*. Princeton University Press, 2009.

[11] J.-M. Dion, C. Commault, and J. V. der Woude, "Generic properties and control of linear structured systems: a survey." *Automatica*, pp. 1125–1144, 2003.

[12] Z. Huang, S. Mitra, and G. Dullerud, "Differentially private iterative synchronous consensus," in *Proceedings of the 2012 ACM Workshop on Privacy in the Electronic Society*, ser. WPES '12. New York, NY, USA: ACM, 2012, pp. 81–90.

[13] M. Kefayati, M. Talebi, B. Khalaj, and H. Rabiee, "Secure consensus averaging in sensor networks using random offsets," *Telecommunications and Malaysia International Conference on Communications, 2007. ICT-MICC 2007. IEEE International Conference on*, pp. 556–560, May 2007.

[14] N. Manitara and C. Hadjicostis, "Privacy-preserving asymptotic average consensus," *2013 European Control Conference (ECC)*, pp. 760–765, July 2013.

[15] Z. Chen, "Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond," McMaster University, Tech. Rep., 2003.

[16] W. Kratz, "Characterization of strong observability and construction of an observer," *Linear Algebra and its Applications*, vol. 221, no. 0, pp. 31 – 40, 1995.

[17] S. Sundaram and C. N. Hadjicostis, "Distributed function calculation via linear iterative strategies in the presence of malicious agents." *IEEE Trans. Automat. Contr.*, vol. 56, no. 7, pp. 1495–1508, 2011.

[18] S. Pequito, *A Structural Approach to Design, Analysis and Optimization of Large-Scale Dynamical Systems*. PhD Dissertation, Carnegie Mellon University and Instituto Superior Tecnico - University of Lisbon, Portugal, 2014.

[19] S. Pequito, S. Kar, and A. Aguiar, "A framework for structural input/output and control configuration selection of large-scale systems," *Submitted to IEEE Transactions on Automatic Control*, 2013. [Online]. Available: http://arxiv.org/pdf/1309.5868v1.pdf

[20] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms*, 2nd ed. McGraw-Hill Higher Education, 2001.

[21] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing." *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1847–1864, 2010.

[22] S. Hosoe, "Determination of generic dimension of controllable subspaces and its applications," *IEEE Transactions on Automatic Control*, vol. 25, pp. 1192–1196, 1980.

[23] C. Commault and J.-M. Dion, "Input addition and leader selection for the controllability of graph-based systems." *Automatica*, vol. 49, no. 11, pp. 3322–3328, 2013.