k-Dimensional Agreement in Multiagent Systems

Gianluca Bianchin, *Member, IEEE*, Miguel Vaquero, Jorge Cortés, *Fellow, IEEE*, and Emiliano Dall'Anese, *Senior Member, IEEE*

***Abstract*—Given a network of agents, we study the problem of designing a distributed algorithm that computes k independent weighted means of the network's initial conditions (namely, the agents agree on a k-dimensional space). Akin to average consensus, this problem finds applications in distributed computing and sensing, where agents seek to simultaneously evaluate k independent functions at a common point by running a single coordination algorithm. We show that linear algorithms can agree on quantities that are oblique projections of the vector of initial conditions, and we provide techniques to design protocols that are compatible with a pre-specified communication graph. More broadly, our results show that a single agreement algorithm can solve k consensus problems simultaneously at a fraction of the complexity of classical approaches but, in general, it requires higher network connectivity.**

***Index Terms*—Consensus algorithms, decentralized control, graph theory, linear time-invariant (LTI) systems, multi-agent systems.**

I. Introduction

Coordination and consensus algorithms are central to many network synchronization problems, including rendezvous, distributed optimization, and distributed computation and sensing. One of the most established coordination algorithms is that of consensus, which can be used to compute asymptotically a *common weighted average* of the agents' initial states—see, for example, the representative works [1], [2], [3]. This work departs from the observation that, in several applications, it is instead of interest to compute *multiple weighted averages* of the initial states, each characterized by a different weighting. Relevant examples of this problem include distributed computation [4] (where agent-specific weights are used to describe heterogeneous computational objectives across agents), task allocation problems [5] (where agent-specific weights are used to model the heterogeneous computational capabilities of the agents), distributed sensing [6], [7] (where agent-specific weights describe heterogeneous accuracies of different sensing devices), and robotic formation [8] (where agent-specific weights allow one to impose agent-specific configurations relative to other agents).

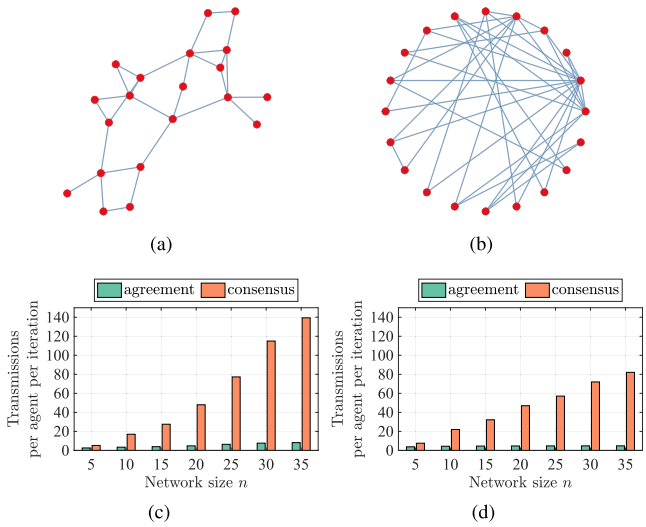

Fig. 1. Communication complexity of running k consensus algorithms in parallel versus one k-dimensional agreement algorithm (proposed in this article) to compute $k = \lfloor \frac{n}{2} \rfloor$ weighted average means of a global quantity. (a) and (c) Erdős–Rényi network model. (b) and (d) Barabasi–Albert model. Bars denote the average number of transmissions per iteration per agent. See Section IV-B.

Mathematically, given a vector $x_0 \in \mathbb{R}^n$ of initial states or estimates–such that each of its entries is known only locally by a single agent–and a rank-k matrix $W \in \mathbb{R}^{n \times n}$, whose rows describe the weights of the means to be computed, we say that the group *reaches a k-dimensional agreement* when, asymptotically, the vector of agents' states converges to Wx_0. The goal of this article is to design distributed control protocols that enable the agents to reach an agreement. A natural approach to tackle this problem consists of executing k consensus algorithms [2] in parallel (see Fig. 1–simulation details are provided in Section III-B), each designed to converge to a specific row of Wx_0. Unfortunately, the communication and computational complexities of such an approach do not scale with the network size (cf., Fig. 1); thus, our objective here is to reach agreements by running *a single* distributed algorithm.

Related work: The problem studied in this work is closely related to that of consensus. Consensus algorithms have been extensively studied in the literature. A list of representative topics (necessarily incomplete) includes: Sufficient and/or necessary conditions for consensus [2], [9], [10], [11], [12], [13], convergence rates [14], [15], and robustness investigations [16]. In contrast with constrained consensus problems [17], [18] (where the agents' states must satisfy agent-dependent constraints during transients, but the desired asymptotic value is unconstrained), in our setting the values are instead constrained at convergence, and thus *the agents' states do not coincide* in general. Clustering-based

Manuscript received 16 March 2024; accepted 2 July 2024. Date of publication 19 July 2024; date of current version 5 December 2024. This work was supported in part by AFOSR Award under Grant FA9550-23-1-0740, in part by projects under Grant PID2022-137909NB-C21 and Grant RED2022-134301-TD, in part by FSR Project SIDDARTA, and in part by NSF Awards CMMI under Grant 2044946 and Grant 2044900. Recommended by Associate Editor Z. Li. *(Corresponding author: Gianluca Bianchin.)*

Gianluca Bianchin is with the ICTEAM Institute and Department of Mathematical Engineering, University of Louvain, 1348 Ottignies-Louvain-la-Neuve, Belgium (e-mail: gianluca.bianchin@uclouvain.be).

Miguel Vaquero is with the School of Science and Technology, IE University, 40003 Segovia, Spain.

Jorge Cortés is with the Department of Mechanical and Aerospace Engineering, University of California San Diego, La Jolla, CA 92093 USA.

Emiliano Dall'Anese is with the Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215 USA.

Digital Object Identifier 10.1109/TAC.2024.3431108

consensus [19], [20], [21] is a closely related problem where the states of agents in the same graph cluster converge to identical values, while intercluster states can differ. Differently from this setting, which is obtained by using weakly connected communication graphs to separate the state of different communities, here we are interested in cases where the asymptotic state of each agent depends on every other agent in the network. To the best of our knowledge, the agreement problem proposed here has not been considered before in the literature. A relevant contribution is that of scaled consensus [22], which can be seen as a special case of the agreement problem studied here, obtained by letting $k = 1$. As shown shortly below, the extension to $k > 1$ is nontrivial as standard assumptions made for consensus are inadequate, see the discussion in Example 4.6.

Contributions: The contribution of this work is threefold. First, we formulate the k-dimensional agreement problem, and we discuss the fundamental limitations of linear protocols in solving this problem. We provide a first main result, which consists of a full characterization of the agreement space for linear protocols. Second, we provide an algebraic characterization of all agreement protocols that are consistent with a pre-specified communication graph. We show how such characterization can be used to design efficient numerical algorithms for agreement. Finally, we illustrate the applicability of the framework on a regression problem through simulations.

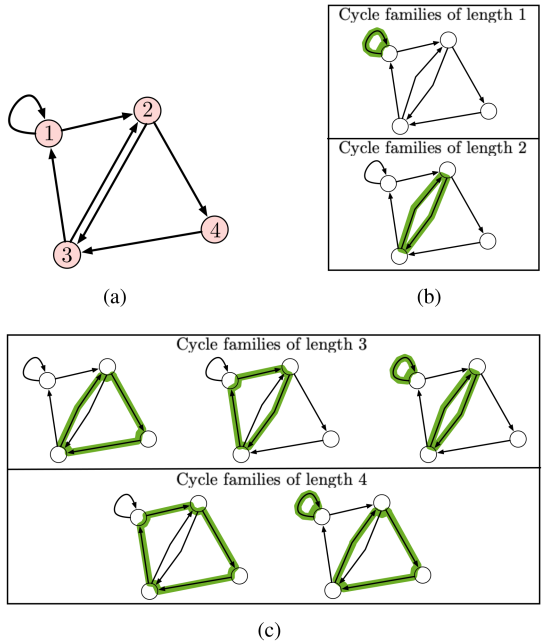

Fig. 2. (a) Example of digraph $\mathcal{G}$. (b) and (c) Illustration of all cycle families of $\mathcal{G}$, organized by length (a cycle family of length ℓ is a set of node-disjoint cycles such that the total number of edges is equal to ℓ).

II. Preliminaries

Notation: $\mathbb{C}$ and $\mathbb{R}$ denote, respectively, the set of complex and real numbers. For $x \in \mathbb{C}$, $\Re(x)$ and $\Im(x)$ denote its real and imaginary parts, respectively. Given $x \in \mathbb{R}^n, u \in \mathbb{R}^m$, $(x, u) \in \mathbb{R}^{n+m}$ denotes their concatenation. $\mathbb{1}_n \in \mathbb{R}^n$ is the vector of all ones, $I_n \in \mathbb{R}^{n\times n}$ is the identity matrix, $\mathbb{0}_{n,m} \in \mathbb{R}^{n\times m}$ is the matrix of all zeros—subscripts are dropped when dimensions are clear from the context. For $A \in \mathbb{R}^{n\times n}$, $\sigma(A) = \{\lambda \in \mathbb{C} \,:\, \det(\lambda I - A) = 0\}$ is its spectrum, and $\lambda_{\max}(A) = \max\{\Re(\lambda) \,:\, \lambda \in \sigma(A)\}$ is its spectral abscissa. For $A \in \mathbb{R}^{n\times m}$, $\mathrm{Im}(A)$ and $\ker(A)$ denote its image and null space, respectively. A polynomial with real coefficients $p(\lambda)$ is *stable* if all its roots have negative real part.

Graph-theoretic notions: A *digraph* is $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, n\}$ and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ are, respectively, the set of nodes and edges. $(i, j) \in \mathcal{E}$ denotes a directed edge from j to i. $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ indicates that $\mathcal{G}$ is a *weighted digraph,* whereby the entries of the *adjacency matrix* $A \in \mathbb{R}^{n\times n}$ describe the edge weights. For $A = [a_{ij}]$ to be a valid adjacency matrix, we must have: $(i, j) \notin \mathcal{E}$ implies $a_{ij} = 0$. If this holds, we say that a matrix A is *consistent* with $\mathcal{G}$. A graph is *complete* if there exists an edge connecting every pair of nodes. A *path* is a sequence of edges $(e_1, e_2, \dots)$, such that the initial node of each edge is the final node of the preceding one. The *length* of a path is the number of edges contained in $(e_1, e_2, \dots)$. A graph is *strongly connected* if, for any $i, j \in \mathcal{V}$, there is a path from i to j. A *closed path* is a path whose initial and final vertices coincide. A closed path is a *cycle* if, going along the path, one reaches no node, other than the initial-final node, more than once. The *length of a cycle* is equal to the number of edges in that cycle. A set of node-disjoint cycles such that the sum of the cycle lengths is equal to ℓ is called a *cycle family* of length ℓ. We let $\mathcal{C}_\ell(\mathcal{G})$ denote the set of all ℓ-long cycle families of $\mathcal{G}$. See Fig. 2 for illustration. Since we are concerned with linear subspaces obtained by forcing certain entries of the matrices in $\mathbb{R}^{n\times n}$ to be zero, we will use the structural approach to system theory [23]. Given $\mathcal{G}$, we let $\mathcal{A}_\mathcal{G} = \{A \in \mathbb{R}^{n\times n} : A \text{ is consistent with } \mathcal{G}\}$ be the vector space of all matrices consistent with $\mathcal{G}$. Let $a \in \mathbb{R}^{|\mathcal{E}|}$, we denote by $\mathbf{A}_\mathcal{G}(a)$ the element of $\mathcal{A}_\mathcal{G}$ parametrized by a.

Projections and linear subspaces: $x, y \in \mathbb{R}^n$ are orthogonal if $x^\mathsf{T} y = 0$; the *orthogonal complement* (or *orthogonal subspace*) of $\mathcal{M} \subset \mathbb{R}^n$ is $\mathcal{M}^\perp := \{x \in \mathbb{R}^n \,:\, x^\mathsf{T} y = 0, \forall\, y \in \mathcal{M}\}$. Given $\mathcal{M}, \mathcal{N} \subseteq \mathbb{R}^n$, $\mathcal{W} \subseteq \mathbb{R}^n$ is a *direct sum* of $\mathcal{M}$ and $\mathcal{N}$ (denoted $\mathcal{W} = \mathcal{M} \oplus \mathcal{N}$) if $\mathcal{M} \cap \mathcal{N} = \{0\}$, and $\mathcal{M} + \mathcal{N} = \{u + v \,:\, u \in \mathcal{M}, v \in \mathcal{N}\} = \mathcal{W}$. Subspaces $\mathcal{M}, \mathcal{N} \subset \mathbb{R}^n$ are *complementary* if $\mathcal{M} \oplus \mathcal{N} = \mathbb{R}^n$. Matrix $\Pi \in \mathbb{R}^{n\times n}$ is called a *projection* if $\Pi^2 = \Pi$. Given complementary subspaces $\mathcal{M}, \mathcal{N} \subset \mathbb{R}^n$, for any $z \in \mathbb{R}^n$ there exists a unique decomposition $z = x + y$, where $x \in \mathcal{M}$, $y \in \mathcal{N}$. The transformation $\Pi_{\mathcal{M},\mathcal{N}}$, defined by $\Pi_{\mathcal{M},\mathcal{N}} z := x$, is called *projection onto $\mathcal{M}$ along $\mathcal{N}$*; $\Pi_{\mathcal{N},\mathcal{M}}$, defined by $\Pi_{\mathcal{N},\mathcal{M}} z := y$, is called *projection onto $\mathcal{N}$ along $\mathcal{M}$*; x is the projection of z onto $\mathcal{M}$ along $\mathcal{N}$, and y is the projection of z onto $\mathcal{N}$ along $\mathcal{M}$. The projection $\Pi_{\mathcal{M},\mathcal{M}^\perp}$ onto $\mathcal{M}$ along $\mathcal{M}^\perp$ is called *orthogonal projection onto $\mathcal{M}$*. Because $\mathcal{M}$ uniquely determines $\mathcal{M}^\perp$, we will denote $\Pi_{\mathcal{M},\mathcal{M}^\perp}$ by $\Pi_\mathcal{M}$. Projections that are not orthogonal are called *oblique projections*.

Lemma 2.1 (See [24, Th. 2.11 and Th. 2.31]): If $\Pi \in \mathbb{R}^{n\times n}$, $\mathrm{rank}(\Pi) = k$, is a projection, there exists $T \in \mathbb{R}^{n\times n}$ such that:

$$\Pi = T \begin{bmatrix} I_k & 0 \\ 0 & 0 \end{bmatrix} T^{-1}.$$

Moreover, if Π is an orthogonal projection, then T can be chosen to be an orthogonal matrix, i.e., $TT^\mathsf{T} = I$. □

Lemma 2.2 (See [24, Th. 2.26]): Let $\mathcal{M}, \mathcal{N}$ be complementary subspaces and let the columns of $M \in \mathbb{R}^{n\times k}$ and $N \in \mathbb{R}^{n\times k}$ form a basis for $\mathcal{M}$ and $\mathcal{N}^\perp$, respectively. Then, $\Pi_{\mathcal{M},\mathcal{N}} = M(N^\mathsf{T} M)^{-1} N^\mathsf{T}$. □

We recall the following known properties [24, Th. 1.60]:

$$\mathrm{Im}(M^\mathsf{T}) = \mathrm{Im}(M^\dagger) = \mathrm{Im}(M^\dagger M) = \mathrm{Im}(M^\mathsf{T} M),$$
$$\ker(M) = \mathrm{Im}(M^\mathsf{T})^\perp = \ker(M^\dagger M) = \mathrm{Im}(I - M^\dagger M).$$

From these properties and Lemma 2.2, if $M \in \mathbb{R}^{m\times n}$, then $\Pi_{\mathrm{Im}(M)} = MM^\dagger$ and $\Pi_{\ker(M)} = I - M^\dagger M$, where $M^\dagger \in \mathbb{R}^{n\times m}$ is the Moore–Penrose inverse of M.

III. Problem Setting

A. Problem Formulation

Consider a set of agents $\mathcal{V} = \{1, \dots, n\}$, each characterized by a state $x_i \in \mathbb{R}, i \in \mathcal{V}$, and communicating through a network whose topology is described by a digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. We study a model where each agent exchanges its state with its neighbors and updates it as

$$\dot{x}_i = a_{ii}x_i + \sum_{j\in\mathcal{N}_i} a_{ij}x_j, \qquad \forall\, i \in \mathcal{V} \tag{1}$$

where $a_{ij} \in \mathbb{R}$, $(i, j) \in \mathcal{E}$, is a weighting factor, and $\mathcal{N}_i = \{j \in \mathcal{V} \setminus \{i\} : (i, j) \in \mathcal{E}\}$ is the set of in-neighbors of i. Setting $A = [a_{ij}], a_{ij} = 0$ if $(i, j) \notin \mathcal{E}$, and $x = (x_1, \dots, x_n)$, in vector form the network dynamics are

$$\dot{x} = Ax. \tag{2}$$

We say that (2) reaches an agreement if each state variable converges to an agent-dependent weighted sum of the initial conditions, as formalized next.

Definition 3.1 (k-dimensional agreement): Let $W \in \mathbb{R}^{n\times n}$ be such that $\mathrm{rank}(W) = k \in \mathbb{N}_{>0}$. We say that the update (2) *globally asymptotically reaches a k-dimensional agreement on* W if, for any $x(0) \in \mathbb{R}^n$

$$\lim_{t\to\infty} x(t) = Wx(0). \tag{3}$$

□

We discuss in Section III-B some application scenarios for this notion. Notice that agreement does not require that the agents' states coincide at convergence: In fact, $\lim_{t\to\infty} \|x_i(t) - x_j(t)\| = 0$ only holds if all rows of W are identical. We discuss in Remark 3.2 how agreement generalizes the well-studied notion of consensus.

Remark 3.2 (Relationship with consensus problems): In the special case $k = 1$, W can be written as $W = vw^\mathsf{T}$ for some $v, w \in \mathbb{R}^n$. In this case, we recover the *scaled consensus* problem [22]. When, in addition, $v = \mathbb{1}$ and $w^\mathsf{T}\mathbb{1} = 1$, we recover the *consensus* problem, see, e.g., [2]. When, $v = \mathbb{1}$ and $w = \frac{1}{n}\mathbb{1}$, our problem simplifies to *average consensus* [2, Sec. 2]. Notice that all state variables converge to the same quantity only when $k = 1$ and $v = \mathbb{1}$. □

In line with the consensus literature, the following distinction is important.

Definition 3.3 (Agreement on some weights vs on arbitrary weights): Let $k \in \mathbb{N}_{>0}$.

a) The set of agents is said to be *globally k-agreement reachable on some weights* if there exists $W \in \mathbb{R}^{n\times n}$, $\mathrm{rank}(W) = k$, and $A \in \mathbb{R}^{n\times n}$ such that (2) globally asymptotically reaches a k-dimensional agreement on W.
b) The set of agents is said to be *globally k-agreement reachable on arbitrary weights* if, for any $W \in \mathbb{R}^{n\times n}$ with $\mathrm{rank}(W) = k$, there exists A such that (2) globally asymptotically reaches a k-dimensional agreement on W. □

Extending Remark 3.2, agreement reachability on some weights is a generalization of global consensus reachability [25], while agreement reachability on arbitrary weights generalizes global *average* consensus reachability [26]. Importantly, whether a group of agents is agreement reachable depends on two main factors: 1) the choice of k, and 2) the connectivity of $\mathcal{G}$. We illustrate this in the following example.

Example 3.4 (Agreement on arbitrary versus on some weights): Consider a set of agents whose communication graph is a set of isolated nodes with self loops (i.e., $\mathcal{V} = \{1, \dots, n\}$ and $\mathcal{E} = \{(i, i)\}_{i\in\mathcal{V}}$). The set of protocols (2) compatible with this graph is characterized by a diagonal matrix $A = \mathrm{diag}(a_1, \dots, a_n)$. Notice that $\lim_{t\to\infty} x(t) = \lim_{t\to\infty} e^{At}x(0)$ exists if and only if $\max\{a_i\}_{i\in\mathcal{V}} \le 0$. When the latter condition holds, $\lim_{t\to\infty} e^{At} = \mathrm{diag}(d_1, \dots, d_n)$, where $d_i = 0$ if $a_i < 0$ and $d_i = 1$ if $a_i = 0$. Hence, the agents are globally agreement reachable on some weights (precisely, any agreement matrix W has the form $W = \mathrm{diag}(d_1, \dots, d_n)$). However, the agents are not globally agreement reachable on arbitrary weights (in fact, agreement cannot be reached, for example, on any nondiagonal W). □

With this motivation, in this work, we study the following two problems.

Problem 1 (Construction of communication graphs for agreement): Determine the largest class of communication graphs that guarantees that the set of agents is globally k-agreement reachable on arbitrary weights. □

Problem 2 (Agreement protocol design): Let $\mathcal{G}$ be a communication graph such that the set of agents is globally k-agreement reachable on arbitrary weights (see Problem 1) and let $W \in \mathbb{R}^{n\times n}, \mathrm{rank}(W) = k$. Determine A, consistent with $\mathcal{G}$, such that (3) holds with optimal rate of convergence. □

Problem 1 is a feasibility problem: It asks to determine the class of graphs that support agreement protocols on arbitrary weights. Problem 2, instead, is a protocol design problem. We conclude this section by discussing an important technical challenge related to designing agreement protocols.

Remark 3.5 (New technical challenges with respect to consensus): Several techniques have been proposed in the literature to design consensus protocols, including Laplacian-based methods [2], distributed weight design [27], and centralized weight design [15]. Most of these methods rely on the assumption that the protocol A is a nonnegative matrix and on the Perron–Frobenius theorem [28] as the main tool for the analysis. Unfortunately, the Perron–Frobenius theorem can no longer be used for agreement problems for two reasons: 1) the entries of W are possibly negative scalars and thus A can no longer be restricted to being a nonnegative matrix; and 2) A can no longer be restricted to being a matrix with a single dominant eigenvalue (as we prove in Lemma 4.1, shortly below). Hence, the agreement problem presents new theoretical challenges with respect to the existing literature. □

B. Illustrative Applications

In this section, we present some illustrative applications where the agreement problem emerges in practice.

Distributed parallel computation of multiple functions: Many numerical computational tasks amount to evaluating a certain function at a given point [29]: Examples include computing scalar addition, inner products, matrix addition and multiplication, matrix powers, finding the least prime factor, etc. [29, Sec. 1.2.3]. Formally, given a function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $(\hat{x}_1, \dots, \hat{x}_n)$, the objective is to evaluate $f(\hat{x}_1, \dots, \hat{x}_n)$. The classical approach to this problem amounts to designing an iterative algorithm $\dot{x} = g(x)$ such that $\lim_{t\to\infty} x(t) = f(\hat{x}_1, \dots, \hat{x}_n)$. When such a computing task has a distributed nature [30], each quantity $\hat{x}_i$ is known only by agent i, and it is of interest to maintain $\hat{x}_i$ private from the rest of the network. In these cases, the distributed computation literature [30] has proposed the update rule $\dot{x}_i = g_i(x)$, to be designed such that $\lim_{t\to\infty} x_i(t) = f(\hat{x}_1, \dots, \hat{x}_n), \forall i$.

Consider now the problem of evaluating, in a distributed fashion, *several* functions at a common point. Formally, given $f_1, \dots, f_n : \mathbb{R}^n \to \mathbb{R}$ and $(\hat{x}_1, \dots, \hat{x}_n)$, the objective is to design distributed protocols of

the form $\dot{x}_i = g_i(x)$ such that

$$\lim_{t\to\infty} x_i(t) = f_i(\hat{x}_1, \dots, \hat{x}_n), \qquad \forall i. \tag{4}$$

It is immediate to see that, when f_i are linear, this is an instance of the agreement problem (3).

Constrained Kalman filtering: Kalman filters are widely used to estimate the states of a dynamic system. In constructing Kalman filters, it is often necessary to account for state-constrained dynamic systems; examples include camera tracking, fault diagnosis, chemical processes, vision-based systems, and biomedical systems [31]. Formally, given a dynamic system of the form $\dot{x} = Fx + Bu + w$, $y = Cx + e$, subject to the state constraint $Dx = 0$, (see [31, eq. (10)]), the objective is that of constructing an optimal estimate $\hat{x}^c$ of x given past measurements $\{y(\tau), \tau \le t\}$. Denoting by $\hat{x}^u$ the state estimate constructed using an unconstrained Kalman filter, a common approach to tackle the constrained problem consists of projecting $\hat{x}^u$ onto the constraint space [31, Sec. 2.3]

$$\hat{x}^c = \arg\min_x \ \|x - \hat{x}^u\|^2, \qquad \text{subject to: } Dx = 0.$$

The solution to this problem is $\hat{x}^c = (I - D^\mathsf{T}(DD^\mathsf{T})^{-1}D)\hat{x}^u$; notice that this is an oblique projection of the vector $\hat{x}^u$. To speed up the calculation, it is often of interest to parallelize the computation of $\hat{x}^c$ across a group of distributed processors. It is then immediate to see that the agreement problem (3) provides a natural framework to address this problem.

C. Complexity Considerations

We now illustrate how the use of classical coordination algorithms to solve (4) leads to a suboptimal use of resources. Assume that functions $f_i(\cdot)$ in (4) are linear, namely, $f_i(x) = w_i^\mathsf{T} x$, with $w_i \in \mathbb{R}^n$, $w_i^\mathsf{T}\mathbb{1} = 1$, and that k vectors of $\{w_1, \dots, w_n\}$ are linearly independent. It is natural to consider two approaches to solve this problem.

Approach 1: This approach consists of running k independent consensus algorithms [26] in parallel, as outlined next. Let each agent i duplicate its state k times: $\{x_i^{(d)} \in \mathbb{R}\}_{d\in\{1,\dots,k\}}$, and update the states using

$$\dot{x}_i^{(d)} = \sum_j a_{ij}^{(d)} \left(x_j^{(d)} - x_i^{(d)}\right), \qquad x_i^{(d)}(0) = \hat{x}_i. \tag{5}$$

Letting $A^{(d)} = [a_{ij}^{(d)}]$ and choosing $A^{(d)}$ such that $w_d^\mathsf{T} A^{(d)} = 0$, (5) is a Laplacian-based consensus algorithm [26, Th. 1]; as such, $\lim_{t\to\infty} x_i^{(d)}(t) = w_d^\mathsf{T}\hat{x}$, provided that $\mathcal{G}$ is strongly connected. In words, the dth state replica of each agent satisfies (4). Unfortunately, the spatial and communication complexities of this approach do not scale well with n (see Fig. 1): Each agent maintains k replica state variables and, at every time step, it transmits these k variables to all its neighbors. Thus, the per-agent spatial complexity is $\mathcal{O}(k)$ (since each agent maintains k state copies), and the per-agent communication complexity[1] is $\mathcal{O}(k \cdot \deg(\mathcal{G}))$ and thus $\mathcal{O}(n \cdot \deg(\mathcal{G}))$ when k grows with n.

Approach 2: Consider the use of protocol (2), designed to achieve (3) with $W = [w_1, \dots, w_n]^\mathsf{T}$. Deriving techniques to design such a protocol is the focus of this work, and will be presented shortly below (see Section V). For such a protocol, the per-agent spatial complexity is $\mathcal{O}(1)$, since each agent maintains a single scalar state variable and the communication complexity is $\mathcal{O}(\deg(\mathcal{G}))$. A comparison of the communication volumes of the two approaches is illustrated in Fig. 1. Notice the fundamental difference between the two approaches: In Approach 1, one computes k independent quantities by *running k distributed averaging algorithms* while, in Approach 2, one computes the k independent quantities by *running a single distributed algorithm*.

[1] $\deg(\mathcal{G})$ denotes the largest among all in- and out-node degrees in $\mathcal{G}$.

IV. Characterization of the Agreement Space and Fundamental Limitations

The focus of this section is to address Problem 1.

A. Algebraic Characterization of Agreement Space

The following result is instrumental.

Lemma 4.1 (Spectral properties of agreement protocols): A set of agents with communication graph $\mathcal{G}$ is globally k-agreement reachable on some weights, if and only if there exists $A \in \mathbb{R}^{n\times n}$ such that

$$A \in \mathcal{A}_\mathcal{G}, \qquad \text{and} \qquad A = T \begin{bmatrix} \mathbb{0}_{k,k} & \mathbb{0}_{k,n-k} \\ \mathbb{0}_{n-k,k} & B \end{bmatrix} T^{-1} \tag{6}$$

for some nonsingular $T \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{(n-k)\times(n-k)}$ satisfying $\lambda_{\max}(B) < 0$.

Conversely, a set of agents is globally k-agreement reachable on arbitrary weights, if and only if for any nonsingular $T \in \mathbb{R}^{n\times n}$, there exists $A \in \mathbb{R}^{n\times n}$ such that (6) holds. □

Proof:(If) When (6) holds, we have that

$$\lim_{t\to\infty} x(t) = \lim_{t\to\infty} e^{At}x(0) = \underbrace{T \begin{bmatrix} I_k & \mathbb{0} \\ \mathbb{0} & \mathbb{0} \end{bmatrix} T^{-1}}_{:=W} x(0) = Wx(0).$$

(Only if): From [28, Lemma 1.7], if $\lim_{t\to\infty} e^{At}$ exists, then $\lambda_{\max}(A) \le 0$; moreover, if λ is an eigenvalue of A such that $\Re(\lambda) = 0$, then $\lambda = 0$ and its algebraic and geometric multiplicities coincide. It follows that A must satisfy (6). ■

Lemma 4.1 provides an algebraic characterization of agreement protocols through (6). Next, we characterize the class of weight matrices W on which an agreement can be reached.

Proposition 4.2 (Characterization of agreement space): Let $x(t)$ denote the solution of (2) with initial condition $x(0)$. If $\lim_{t\to\infty} x(t) := x_\infty$ exists, then there exist complementary subspaces $\mathcal{M}, \mathcal{N} \subset \mathbb{R}^n$ such that $x_\infty = \Pi_{\mathcal{M},\mathcal{N}} x(0)$. Moreover, let $\{t_1, \dots, t_k\}$ denote the first k columns of T in (6) and $\{\tau_1^\mathsf{T}, \dots, \tau_k^\mathsf{T}\}$ denote the first k rows of T^{-1}. Then

$$\mathcal{M} = \mathrm{Im}(\{t_1, \dots, t_k\}), \qquad \mathcal{N}^\perp = \mathrm{Im}(\{\tau_1, \dots, \tau_k\}). \tag{7}$$

□

The proof of this claim is available in [32].

Proposition 4.2 is a fundamental limitation-type result: It shows that if $\dot{x} = Ax$ converges, then the asymptotic value is some oblique projection of the initial conditions $x(0)$. In turn, this implies that linear protocols can agree only on weight matrices W that are oblique projections.

Remark 4.3 (Geometric reinterpretation of consensus algorithms): In the case of consensus, the group of agents is known to converge to $\mathbb{1}w^\mathsf{T}x(0)$, where w is the left eigenvector of A that satisfies $w^\mathsf{T}\mathbb{1} = 1$ (see Remark 3.2). Proposition 4.2 allows us to give a geometric interpretation of the consensus value: $\mathbb{1}w^\mathsf{T}x(0) = \Pi_{\mathcal{M},\mathcal{N}}x(0)$ is the oblique projection of $x(0)$ onto $\mathcal{M} = \mathrm{Im}(\mathbb{1})$ along $\mathcal{N} = \mathrm{Im}(w)^\perp$. In the case of average consensus, the convergence value (given by $\frac{1}{n}\mathbb{1}\mathbb{1}^\mathsf{T}x(0)$) is the orthogonal projection of $x(0)$ onto $\mathcal{M} = \mathrm{Im}(\mathbb{1})$. □

Motivated by the conclusions in Proposition 4.2, in what follows we make the following assumption.

Algorithm 1: Construction of agreement matrix A.

Require $M \in \mathbb{R}^{n\times k}$ whose columns form a basis for $\mathcal{M}$
Require $N \in \mathbb{R}^{n\times k}$ whose columns form a basis for $\mathcal{N}^\perp$
$\Pi_{\mathcal{M},\mathcal{N}} \leftarrow M(N^\mathsf{T}M)^{-1}N^\mathsf{T}$;
Determine T such that $\Pi_{\mathcal{M},\mathcal{N}} = T\begin{bmatrix} I_k & \mathbb{0} \\ \mathbb{0} & \mathbb{0}\end{bmatrix}T^{-1}$;
Choose $B \in \mathbb{R}^{(n-k)\times(n-k)}$ such that $\lambda_{\max}(B) < 0$;
return $A = T\begin{bmatrix} \mathbb{0}_k & \mathbb{0} \\ \mathbb{0} & B\end{bmatrix}T^{-1}$;

Assumption 1 (Matrix of weights is a projection): The matrix of weights W is a projection. Namely, $W \in \mathbb{R}^{n\times n}$, $W^2 = W$, and $\operatorname{rank}(W) = k$. □

Notice that, given two complementary subspaces $\mathcal{M},\mathcal{N}$, a matrix W that satisfies Assumption 1 can be computed as

$$W = M(N^\mathsf{T}M)^{-1}N^\mathsf{T}$$

where $M \in \mathbb{R}^{n\times k}$ and $N \in \mathbb{R}^{n\times k}$ form a basis for $\mathcal{M}$ and $\mathcal{N}^\perp$, respectively (see Lemma 2.2). Notice that the agreement space corresponding to this choice of W is $\Pi_{\mathcal{M},\mathcal{N}}$.

We are now ready to prove the following.

Proposition 4.4 (Existence of agreement algorithms over complete digraphs): Let $\mathcal{M},\mathcal{N} \subset \mathbb{R}^n$ be complementary subspaces and $\mathcal{G}$ the complete graph. There exists $A \in \mathbb{R}^{n\times n}$, consistent with $\mathcal{G}$, such that the iterates (2) satisfy $\lim_{t\to\infty} x(t) = \Pi_{\mathcal{M},\mathcal{N}}x(0)$. □

Proof: For any pair of complementary subspaces $\mathcal{M},\mathcal{N}$, [24, Th. 2.26] guarantees the existence of an oblique projection matrix $\Pi_{\mathcal{M},\mathcal{N}}$. Moreover, by Lemma 2.1, there exists invertible $T_\Pi \in \mathbb{R}^{n\times n}$ such that $\Pi_{\mathcal{M},\mathcal{N}}$ can be decomposed as

$$\Pi_{\mathcal{M},\mathcal{N}} = T_\Pi \begin{bmatrix} I_k & \mathbb{0} \\ \mathbb{0} & \mathbb{0}\end{bmatrix} T_\Pi^{-1}$$

where $k = \dim(\mathcal{M})$. The statement follows by choosing A as in (6) with $T = T_\Pi$ and by noting that, with this choice, $\lim_{t\to\infty} e^{At}x(0) = \Pi_{\mathcal{M},\mathcal{N}}x(0)$. ■

Proposition 4.4 provides a preliminary answer to Problem 1: if the communication graph is complete, a set of agents is globally k-agreement reachable on arbitrary weights, $\forall\, k \in \mathbb{N}_{>0}$. The proof is constructive, and it provides a way to derive agreement protocols—see Algorithm 1. We remark that, for some special choices of $\mathcal{M},\mathcal{N}$, one or more entries of A may be identically zero (notice that such A remain consistent with our definition of adjacency matrix for complete graphs—see Section II); in these cases, the protocol A could also be implemented over a noncomplete graph. However, in the general case, A is nonsparse.

B. Structural Necessary Conditions for Agreement

While Proposition 4.4 shows that complete graphs can reach an agreement on arbitrary weights, it remains unclear whether this property also holds for graphs with weaker connectivity. We begin by showing that strong connectivity[2] is necessary but not sufficient for agreement reachability on arbitrary weights.

Lemma 4.5 (Necessity of strong connectivity): A set of agents is globally k-agreement reachable on arbitrary weights only if $\mathcal{G}$ is strongly connected. □

[2]Recall that strong connectivity is necessary and sufficient for global *average* consensus reachability [26].

Proof: When $\mathcal{G}$ is not strongly connected, for all A consistent with $\mathcal{G}$, at least one of the entries of $\lim_{t\to\infty} e^{At}$ is identically zero (this follows from $e^{At} = \sum_{i=0}^{\infty} \frac{A^i t^i}{i!}$ and [28, Cor 4.5]). In this case, since $W = \lim_{t\to\infty} e^{At}$, an agreement cannot be reached on every W such that $w_{ij} \neq 0\ \forall\, i,j$.. ■

Example 4.6 (Strong connectivity is not sufficient for agreement on arbitrary weights): Assume that a network of $n = 3$ agents is interested in agreeing on a space with $k = 2$ by using a noncomplete communication graph $\mathcal{G}$. By using Lemma 4.1, the agents are agreement reachable on arbitrary weights only if

$$A = \underbrace{\begin{bmatrix} t_1 & t_2 & t_3\end{bmatrix}}_{=T} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \beta\end{bmatrix} \underbrace{\begin{bmatrix} \tau_1 & \tau_2 & \tau_3\end{bmatrix}^\mathsf{T}}_{=T^{-1}} = \beta t_3 \tau_3^\mathsf{T} \tag{8}$$

for some β such that $\Re(\beta) < 0$ and some $T \in \mathbb{R}^{3\times 3}$. By (8), A must be a rank-one matrix and, since $\mathcal{G}$ is not complete, at least one of the entries of A must be identically zero. These two properties imply that at least one of the rows or columns of A must be identically zero, and thus that $\mathcal{G}$ cannot be strongly connected. Since $\mathcal{G}$ is not strongly connected, by [28, Cor 4.5] at least one of the rows or columns of $W = \lim_{t\to\infty} e^{At}$ must be identically zero. In summary, we have found that the agents are globally two-agreement reachable on arbitrary weights only if $\mathcal{G}$ is the complete graph. □

We will thus make the following necessary assumption.

Assumption 2 (Strong connectivity): The communication digraph $\mathcal{G}$ is strongly connected. □

V. Agreement Algorithms Over Sparse Digraphs

While (6) gives a full characterization of agreement protocols and can be used to design agreement algorithms over complete graphs (cf., Algorithm 1), it remains unclear how to design agreement protocols when $\mathcal{G}$ is not complete. This is the focus of this section. We will often use the following decomposition for W (see Assumption 1 and Lemma 2.1):

$$W = T\begin{bmatrix} I_k & \mathbb{0} \\ \mathbb{0} & \mathbb{0}\end{bmatrix} T^{-1} \tag{9}$$

where $T \in \mathbb{R}^{n\times n}$ is invertible. Moreover, we will use

$$T = \begin{bmatrix} t_1 & \cdots & t_n\end{bmatrix}, \qquad T^{-1} = \begin{bmatrix} \tau_1 & \cdots & \tau_n\end{bmatrix}^\mathsf{T} \tag{10}$$

where $t_i, \tau_i \in \mathbb{R}^n, i \in \{1,\dots,n\}$ (notice that $\tau_i^\mathsf{T}t_j = 1$ if $i = j$ and $\tau_i^\mathsf{T}t_j = 0$ otherwise).

A. Algebraic Conditions for Sparse Digraphs

We will use a graph-theoretic interpretation of characteristic polynomials [33], which we recall next. Recall that $\mathcal{C}_\ell(\mathcal{G})$ denotes the set of all ℓ-long cycle families of $\mathcal{G}$ (see Section II).

Lemma 5.1 ([33, Th. 1]): Let $\mathcal{G}$ be a digraph, let $A \in \mathcal{A}_\mathcal{G}$, and $\det(\lambda I - A) = \lambda^n + p_1\lambda^{n-1} + \cdots + p_{n-1}\lambda + p_n$ be its characteristic polynomial. Then, for all $p_\ell, \ell \in \{1,\dots,n\}$

$$p_\ell = \sum_{\xi\in\mathcal{C}_\ell(\mathcal{G})} (-1)^{d(\xi)} \prod_{(i,j)\in\xi} a_{ij}$$

where $d(\xi)$ is the number of cycles in cycle family ξ. □

The lemma provides a graph-theoretic description of the characteristic polynomial: It shows that the ℓth coefficient of $\det(\lambda I - A)$ is a sum of terms; each summand is the product of edges in a cycle family of length ℓ of $\mathcal{G}$.

Example 5.2: Consider the digraph in Fig. 2(a). We have

$$\mathbf{A}_{\mathcal{G}}(a) = \begin{bmatrix} a_{11} & 0 & a_{13} & 0 \\ a_{21} & 0 & a_{23} & 0 \\ 0 & a_{32} & 0 & a_{34} \\ 0 & a_{42} & 0 & 0 \end{bmatrix}$$

and we refer to Fig. 2(b) and (c) for an illustration of all cycle families of this graph. Lemma 5.1 yields

$$p_1 = -a_{11}, \qquad p_3 = -a_{13}a_{21}a_{32} + a_{11}a_{23}a_{32} - a_{23}a_{42}a_{34},$$
$$p_2 = -a_{23}a_{32} \qquad p_4 = -a_{13}a_{21}a_{42}a_{34} + a_{11}a_{23}a_{34}a_{42}.$$

Notice that each summand in p_ℓ is the product of weights in a cycle family of the corresponding size [cf., Fig. 2(b) and (c)]. □

The following result is one of the main contributions of this article. Recall that for $a \in \mathbb{R}^{|\mathcal{E}|}$, $\mathbf{A}_{\mathcal{G}}(a)$ is the matrix consistent with $\mathcal{G}$ whose entries are parametrized by a.

Theorem 5.3 (Algebraic characterization of sparse agreement matrices): Let Assumptions 1 and 2 hold. $\dot{x} = \mathbf{A}_{\mathcal{G}}(a)x$ globally asymptotically reaches a k-dimensional agreement on W if and only if the following hold simultaneously:

a) $\mathbf{A}_{\mathcal{G}}(a)t_i = 0, \quad \tau_i^{\mathsf{T}}\mathbf{A}_{\mathcal{G}}(a) = 0, \qquad \forall i \in \{1, \dots, k\}$;

b) The polynomial $\lambda^{n-k-1} + p_1\lambda^{n-k-2} + \cdots + p_{n-k-1}$, whose coefficients are defined as

$$p_\ell = \sum_{\xi \in \mathcal{C}_\ell(\mathcal{G})} (-1)^{d(\xi)} \prod_{(i,j)\in\xi} a_{ij}, \qquad \ell \in \{1, \dots, n-k\}$$

is stable. □

Proof: (If) Let A be any matrix that satisfies *a)–b)*. If A is diagonalizable, then, by letting $T = (t_1, \dots, t_n)$ be the matrix of its right eigenvectors and $(T^{-1})^{\mathsf{T}} = (\tau_1, \dots, \tau_n)$ be the matrix of its left eigenvectors, we conclude that A satisfies (6) and thus the linear update reaches an agreement on W. If A is not diagonalizable, let T be a similarity transformation such that $T^{-1}AT$ is in Jordan normal form

$$T^{-1}AT = \begin{bmatrix} J_{\lambda_1} & & & \\ & J_{\lambda_2} & & \\ & & \ddots & \\ & & & J_{\lambda_n} \end{bmatrix}, \quad J_{\lambda_i} = \begin{bmatrix} \lambda_1 & 1 & & \\ & \ddots & \ddots & \\ & & & \lambda_1 \end{bmatrix}.$$

From *a)* we conclude that $\lambda = 0$ is an eigenvalue with algebraic multiplicity k, moreover, since the vectors t_i are linearly independent (see (9)), we conclude that its geometric multiplicity is also equal to k, and thus all Jordan blocks associated with $\lambda = 0$ have dimension 1. Namely, $J_{\lambda_1} = \cdots = J_{\lambda_k} = 0$. By combining this with *b)*, we conclude that the characteristic polynomial of A is

$$\det(\lambda I - A) = \lambda^n + p_1\lambda^{n-1} + \cdots + p_{n-k-1}\lambda^{k-1}$$

and, since by assumption such polynomial is stable, we conclude that all remaining eigenvalues $\{\lambda_{k+1}, \dots, \lambda_n\}$ of A satisfy $\Re(\lambda_i) < 0$. Since all Jordan blocks associated with $\lambda = 0$ have dimension 1 and all the remaining eigenvalues of A are stable, we conclude that A admits the representation (6) and thus the linear update reaches an agreement.

(Only if): We will prove this claim by showing that (6) implies *a)–b)*. To prove that *a)* holds, we rewrite (6) as

$$T^{-1}AT = \begin{bmatrix} \mathbb{0} & \mathbb{0} \\ \mathbb{0} & B \end{bmatrix}$$

and, by taking the first k columns of the above identity we conclude $At_i = 0$, $i \in \{1, \dots, k\}$, thus showing that *a)* holds. To prove that *b)* holds, notice that (6) implies that the characteristic polynomial of A is a stable polynomial with k roots at zero. Namely

$$\det(\lambda I - A) = \lambda^k(\lambda - \lambda_1)(\lambda - \lambda_2)\cdots(\lambda - \lambda_{n-k})$$
$$= \lambda^n + p_1\lambda^{n-1} + \cdots + p_{n-k-1}\lambda^{k-1}$$

where $\Re(\lambda_i) < 0, i \in \{1, \dots, n-k\}$ and $p_j, j \in \{1, \dots, n-k-1\}$, are nonzero real coefficients. The statement *b)* thus follows by applying the graph-theoretic interpretation of the coefficients of the characteristic polynomial in Lemma 5.1. ■

Theorem 5.3 provides an algebraic characterization of agreement protocols over sparse digraphs. The result is remarkable as it can be used to design sparse agreement protocols as follows. Given $\mathcal{G}$ and W, we interpret a as well as $p_1, \dots p_{n-k}$ as free parameters or unknowns; then, *a)–b)* define a system of equations (precisely, $2nk$ linear equations and $n-k$ multilinear polynomial equations) in these unknowns. Any solution to this system of equations–yielding a stable characteristic polynomial–gives an agreement protocol on W consistent with $\mathcal{G}$. Notice that the solvability of these equations is not guaranteed in general, but it can be assessed via standard techniques, as discussed in the following remark.

Remark 5.4 (Determining solutions to systems of polynomial equations): A powerful technique for determining solutions to systems of polynomial equations uses the tool of Gröbner bases, as applied using Buchberger's algorithm [34]. The technique relies on transforming the system of equations into a canonical form, expressed in terms of a Gröbner basis, for which it is then easier to determine a solution. We refer to [34] and [35] for a complete discussion. Furthermore, existence of solutions can be assessed using Hilbert's Nullstellensatz theorem [35]. In short, the theorem guarantees that a system of polynomial equations has no solution, if and only if its Gröbner basis is $\{1\}$. In this sense, the Gröbner basis method provides an easy way to check solvability of *a)–b)*. We also note that the computational complexity of solving systems of polynomial equations via Gröbner bases is exponential [35]. □

B. Fast Distributed Agreement Algorithms

We next tackle Problem 2. The freedom in the choice of $p_1, \dots, p_{n-k}$ in Theorem 5.3 suggests that a certain graph may admit multiple consistent agreement protocols. We will now leverage such freedom to seek protocols with optimal rate of convergence. Problem 2 can be made formal as follows:

$$\min_A \; r(A)$$
$$\text{s.t. } A \in \mathcal{A}_{\mathcal{G}}, \quad \lim_{t\to\infty} e^{At} = W. \tag{11}$$

In (11), $r : \mathbb{R}^{n\times n} \to \mathbb{R}$ is a function that measures the rate of convergence of e^{At}. By Lemma 4.1, the optimization (11) is feasible, if and only if (6) holds with T given by (9).

When the optimization problem (11) is feasible, it is natural to consider two possible choices for the cost function $r(\cdot)$, motivated by the size of $\|e^{At}\|$ as a function of time. The first limiting case is $t \to \infty$. In this case, we consider the following asymptotic measure of convergence motivated by [36, Ch. 14]:

$$r_\infty(A) := \lim_{t\to\infty} t^{-1} \log \|e^{At}\| = \lambda_{\max}(A) \tag{12}$$

where $\lambda_{\max}(A)$ is the spectral abscissa of A (see Section II). The second limiting case is $t \to 0$. In this case

$$r_0(A) := \frac{d}{dt}\|e^{At}\|\Big|_{t=0} = \lim_{t\downarrow 0} t^{-1} \log \|e^{At}\| = \lambda_{\max}\left(\frac{A + A^{\mathsf{T}}}{2}\right) \tag{13}$$

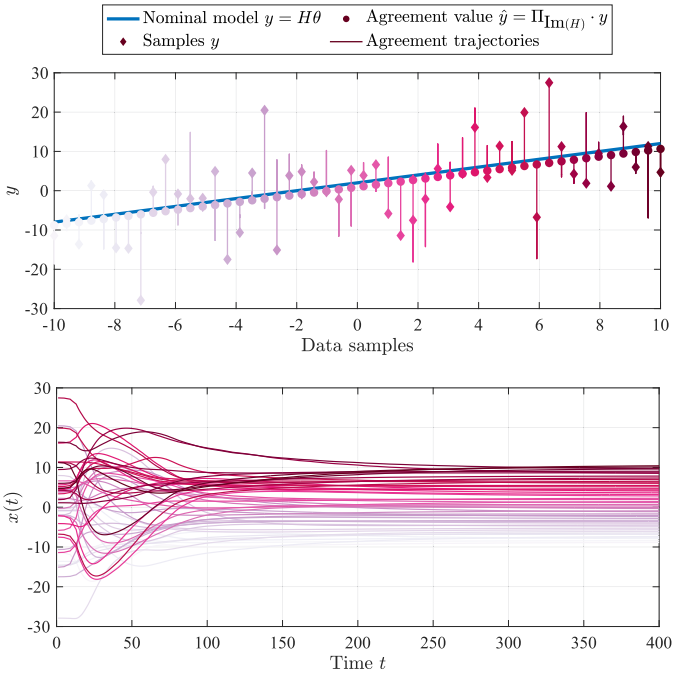

Fig. 3. Application of the agreement problem to solve a regression problem. Each agent can measure a sample y_i (represented by diamond markers) and cooperatively computes the projection of $\hat{y}_i$ onto the range of the regression matrix H. (Top figure) continuous lines illustrate the time evolution of the states of (1). (Bottom figure) Time evolution of the trajectories of (1). Notice that the agents' states do not converge to the same value.

where $\lambda_{\max}(\frac{A+A^{\mathsf{T}}}{2})$ is the numerical abscissa of A [36].

We have the following result.

Proposition 5.5 (Fast agreement problem): Let Assumptions 1 and 2 hold. Assume that the optimization problem (11) is feasible. Any solution to the following optimization problem:

$$\min_{a\in\mathbb{R}^{|\mathcal{E}|}} \; r(\mathbf{A}_{\mathcal{G}}(a))$$
$$\text{s.t. } \mathbf{A}_{\mathcal{G}}(a)t_i = 0,\ \tau_i^{\mathsf{T}}\mathbf{A}_{\mathcal{G}}(a) = 0, \qquad i \in \{1,\dots,k\} \tag{14}$$

where t_i, τ_i are as in (10), is also a solution of (11). □

The proof of this claim is available in [32].

Proposition 5.5 allows us to recast (11) as a finite-dimensional search over the parameters $a \in \mathbb{R}^{|\mathcal{E}|}$. We remark that (14) with the numerical abscissa formulation (13) is a convex optimization problem [37], while with the spectral abscissa formulation (12), finding solutions may be computationally burdensome because the objective function may be nonconvex (or even nonLipschitz [37]).

VI. Applications and Numerical Validation

Consider a distributed estimation problem characterized by a regression model of the form $y = H\theta + w$, where $H \in \mathbb{R}^{n\times k}, n > k, \theta \in \mathbb{R}^k$ is an unknown parameter, and $w \in \mathbb{R}^n$ models noise. We assume that each agent i can sense the ith entry of vector y, denoted by y_i, and the group of agents is interested in cooperatively solving the regression problem

$$\theta_{\text{ls}} := \arg\min_{\theta} \|H\theta - y\|. \tag{15}$$

It is well-known that θ_{ls} is given by $\theta_{\text{ls}} = (H^{\mathsf{T}}H)^{-1}H^{\mathsf{T}}y$, provided that $H^{\mathsf{T}}H$ is invertible. Thus, the vector to be computed by the agents (denoised measurements) is

$$\hat{y} = H\theta_{\text{ls}} = H(H^{\mathsf{T}}H)^{-1}H^{\mathsf{T}}y$$

which is the orthogonal projection of y onto $\mathrm{Im}(H)$. For figure illustration purposes, we consider the case $n = 50$ (meaning $n = 50$ agents or sensors in the network) and $k = 2$ (meaning the sensor measurements is interpolated using a line). We computed an agreement protocol using the optimization problem (13) and (14) with weights $W = H(H^{\mathsf{T}}H)^{-1}H^{\mathsf{T}}$ and implemented on a circulant graph [2], where each agent communicates with its four nearest neighbors. Fig. 3(top) shows the sampling points y and asymptotic estimates $\hat{y}$ in comparison with the true regression model. As expected, (1) converges to the data points corresponding to the Mean Square Error Estimator. Fig. 3(bottom) shows the trajectories of the agents' states. Notice that the agreement state is a 50-D vector constrained to a 2-D subspace.

VII. Conclusion

We studied the k-dimensional agreement problem, whereby a group of agents seeks to compute k independent weighted means of the agents' initial states. We provided algebraic conditions to check the feasibility of the problem and algorithms to design such protocols. Our results show that agreement protocols can compute several weighted means of the agents' initial conditions at a fraction of the complexity of existing consensus algorithms. This work opens the opportunity for multiple directions of future research; among them, we highlight the derivation of graph-theoretic conditions to solve Problem 1, the design of agreement protocols in a distributed way, the use of nonlinear dynamics, and the synthesis of distributed protocols to solve optimization problems over networks.

References

[1] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," in *Proc. IEEE Conf. Decis. Control*, 2005, pp. 2996–3000.

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

[3] W. Ren, R. W. Beard, and E. M. Atkins, "A survey of consensus problems in multi-agent coordination," in *Proc. Amer. Control Conf.*, Portland, OR, USA, Jun. 2005, pp. 1859–1864.

[4] D. Peleg, *Distributed Computing: A Locality-Sensitive Approach*. Philadelphia, PA, USA: SIAM, 2000.

[5] H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," *IEEE Trans. Robot.*, vol. 25, no. 4, pp. 912–926, Aug. 2009.

[6] L. Zhao and A. Abur, "Multi area state estimation using synchronized phasor measurements," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 611–617, May 2005.

[7] F. Pasqualetti, R. Carli, A. Bicchi, and F. Bullo, "Distributed estimation and detection under local information," in *Proc. IFAC Workshop Distrib. Estimation Control Networked Syst.*, Annecy, France, Sep. 2010, pp. 263–268.

[8] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[9] J. N. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, Nov. 1984. [Online]. Available: http://web.mit.edu/jnt/www/Papers/PhD-84-jnt.pdf

[10] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.

[11] M. Cao, A. S. Morse, and B. D. O. Anderson, "Reaching a consensus in a dynamically changing environment - convergence rates, measurement delays and asynchronous events," *SIAM J. Control Optim.*, vol. 47, no. 2, pp. 601–623, 2008.

[12] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 655–661, May 2005.
[13] J. M. Hendrickx and J. N. Tsitsiklis, "Convergence of type-symmetric and cut-balanced consensus seeking systems," *IEEE Trans. Autom. Control*, vol. 58, no. 1, pp. 214–218, Jan. 2013.
[14] A. Olshevsky and J. N. Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM J. Control Optim.*, vol. 48, no. 1, pp. 33–55, 2009.
[15] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, pp. 65–78, 2004.
[16] D. Dolev, N. A. Lynch, S. Pinter, E. W. Stark, and W. E. Weihl, "Reaching approximate agreement in the presence of faults," *J. ACM*, vol. 33, no. 3, pp. 499–516, 1986.
[17] A. Nedić, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 922–938, Apr. 2010.
[18] F. Morbidi, "Subspace projectors for state-constrained multi-robot consensus," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 7705–7711.
[19] S. Ahmadizadeh, I. Shames, S. Martin, and D. Nešić, "On eigenvalues of Laplacian matrix for a class of directed signed graphs," *Linear Algebra Appl.*, vol. 523, pp. 281–306, 2017.
[20] W. Li and H. Dai, "Cluster-based distributed consensus," *IEEE Trans. Wireless Commun.*, vol. 8, no. 1, pp. 28–31, Jan. 2009.
[21] G. Bianchin, A. Cenedese, M. Luvisotto, and G. Michieletto, "Distributed fault detection in sensor networks via clustering and consensus," in *Proc. IEEE Conf. Decis. Control*, Osaka, Japan, Dec. 2015, pp. 3828–3833.
[22] S. Roy, "Scaled consensus," *Automatica*, vol. 51, pp. 259–262, 2015.
[23] K. Röbenack and K. J. Reinschke, "Digraph based determination of Jordan block size structure of singular matrix pencils," *Linear Algebra Appl.*, vol. 275/276, pp. 495–507, 1998.
[24] A. Galántai, *Projectors and Projection Methods*. Berlin, Germany: Springer, 2003, vol. 6.
[25] R. W. Beard and V. Stepanyan, "Information consensus in distributed multiple vehicle coordinated control," in *Proc. IEEE Conf. Decis. Control*, vol. 2, 2003, pp. 2029–2034.
[26] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.
[27] V. Schwarz, G. Hannak, and G. Matz, "On the convergence of average consensus with generalized metropolis-hasting weights," in *Proc. EEE Conf. Acoust., Speech Signal Process.*, 2014, pp. 5442–5446.
[28] F. Bullo, *Lectures on Network Systems*. Santa Barbara, CA, USA: Kindle Direct Publishing, 2019, vol. 1.
[29] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Nashua, NH, USA: Athena Sci., 2015.
[30] S. Micali and P. Rogaway, *Secure Computation*. Berlin, Germany: Springer, 1992.
[31] D. Simon, "Kalman filtering with state constraints: A survey of linear and nonlinear algorithms," *IET Control Theory Appl.*, vol. 4, no. 8, pp. 1303–1318, 2010.
[32] G. Bianchin, M. Vaquero, J. Cortés, and E. Dall'Anese, "k-dimensional agreement in multiagent systems for distributed coordination," Aug. 2022, *arXiv:2208.08999*.
[33] K. J. Reinschke, "Graph-theoretic approach to symbolic analysis of linear descriptor systems," *Linear Algebra Appl.*, vol. 197, pp. 217–244, 1994.
[34] D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. Berlin, Germany: Springer, 2013.
[35] M. Kreuzer and L. Robbiano, *Computational Commutative Algebra*. Berlin, Germany: Springer, 2000, vol. 1.
[36] L. N. Trefethen and M. Embree, *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*. Princeton, NJ, USA: Princeton Univ. Press, 2005.
[37] J. V. Burke and M. L. Overton, "Variational analysis of non-Lipschitz spectral functions," *Math. Program.*, vol. 90, no. 2, pp. 317–351, 2001.