ON THE LIMITS OF FINITE-TIME DISTRIBUTED CONSENSUS THROUGH SUCCESSIVE LOCAL LINEAR OPERATIONS

Mario Coutino[†‡], *Elvin Isufi*[†], *Takanori Maehara*[‡], *Geert Leus*[†]

[†]Delft University of Technology, Delft, The Netherlands
[‡]AIP RIKEN, Tokyo, Japan

ABSTRACT

In this work, we explore the limits of finite-time distributed consensus through the intersection of graph filters and matrix function theory. We focus on algorithms capable to compute the consensus exactly through filtering operations over a graph, and that have been proven to converge in finite time. In this context, we show that there exists an algebraic algorithm that can minimize the minimum polynomial of a matrix whose support is known. Different from previous works, we leverage the structure of matrices that share the same support and are diagonalizable by the eigenbasis of the graph shift operator to prove a theoretical result with respect to the minimum number of diffusion steps required to reach consensus. We show that the previously known bound on the number of consensus iterations can be further reduced in accordance to the algebraic properties of the matrix representation of the network. Finally, insights with respect to the relation between the graph topology and the algebraic properties of such matrices are provided in order to encourage further discussion on the role of eigenvalues and eigenvectors in the network topology.

***Index Terms*—** Consensus, distributed averaging, graph filters, signal processing over networks

1. INTRODUCTION

Since its first appearance [1], distributed consensus has been a subject of extensive research [2–8]. Most of these works aim at addressing the design question: *which combining rule of local information provides exact/approximate consensus in a distributed network of agents?*

This question can be mathematically expressed by first considering a network with n agents, whose communication capabilities are described by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Here, $\mathcal{V} = \{v_1, \dots, v_n\}$ denotes the set of agents (nodes) and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ the set of pair-wise communication links (edges). For the ith agent, we define its reachable set $\mathcal{N}_i = \{j \,|\, (v_i, v_j) \in \mathcal{E}, j \neq i\}$ as the set of agents that agent v_i can communicate with. Under this model, each agent v_i at time k is assumed to hold a (scalar or vector) quantity $x_i(k)$. Agents share their information with the members of their reachable set and apply a combining rule to the received information to update their values as

$$x_i(k+1) = f_k(x_i(k), \{x_j(k)\}_{j\in\mathcal{N}_i}), \tag{1}$$

where $f_k(\cdot)$ is the *combining rule* at time instant k. Therefore, the design problem for achieving exact/approximate consensus is reduced to that of finding a suitable $f_k(\cdot)$ for all nodes to reach the value

$$y = \frac{1}{N}\Big(x_1(0) + x_2(0) + \cdots + x_n(0)\Big), \tag{2}$$

in the minimum number of communication rounds.

Typically, the combining rule is chosen to be linear, i.e.,

$$x_i(k+1) = w_{ii}(k)x_i(k) + \sum_{j\in\mathcal{N}_i} w_{ij}(k)x_j(k), \tag{3}$$

and the design problem is reduced to only finding the (possibly) time-varying weights $\{w_{ij}(k)\}$ to achieve consensus in the network.

By using linear combining rules or their equivalent formulations using successive products, guarantees for finite-time consensus have been already obtained. In [9], a connection between finite-time consensus and an autoregressive relation on the graph signal, $x_i(k)$, has been established. Using such a relation, it has been shown that consensus on a network can be achieved in a number of time instances equal to the order of a specific matrix polynomial related to the matrix constructed using the weights in (3). Further, the authors in [10] proposed a framework using graph filters [11–13] to provide guarantees for finite-time consensus on arbitrary networks. In the same context, in [14] and [15] it is shown that finite-time consensus can be achieved using the so-called node-variant and edge-variant graph filters.

Although all these works draw tools from different fields and their construction seems to be specific to their particular research area, in this work, we provide a unified view of these methods through the framework of matrix function theory. This framework not only provides a natural way to derive such expressions, but also gives insights into how to further improve them.

In this context, we provide a constructive method for the existence of an algorithm to design matrices with fixed support for achieving consensus in the fastest possible way. We do so by optimizing an algebraic property of matrices with fixed support. Hence, this approach, up to the best of our knowledge, is the most general formulation for distributed consensus though restricted to matrix functions. Furthermore, as we recognize that in many cases this algebraic construction might prove itself difficult, we show that if we restrict ourselves to matrices not only sharing support but also eigenbasis, we can still improve current state-of-the-art results for finite-time consensus. This result also provides new insights into the relation between the eigenvectors and eigenvalues of the graph Laplacian and the topological structure of the related graph.

2. ON THE HERMITE INTERPOLATING POLYNOMIAL

To initiate a general discussion on finite-time consensus through successive local linear operations, we need to draw relations with

This research is supported in part by the ASPIRE project (project 14926 within the STW OTP programme), financed by the Netherlands Organization for Scientific Research (NWO). Mario Coutino is partially supported by CONACYT and AIP RIKEN.

the theory of matrix functions [16]. To do so, we use concepts from polynomials with matrix arguments and the definition of matrix functions via a scalar function $f(\cdot)$ and the Jordan canonical form of $\boldsymbol{A}$, i.e.,

$$f(\boldsymbol{A}) \triangleq \boldsymbol{Z}\text{diag}(f(\boldsymbol{J}_1),\ldots,f(\boldsymbol{J}_s))\boldsymbol{Z}^{-1}, \tag{4}$$

where

$$f(\boldsymbol{J}_k) \triangleq \begin{bmatrix} f(\lambda_k) & f^{(1)}(\lambda_k) & \cdots & \frac{f^{(n_k-1)}(\lambda_k)}{(m_k-1)!} \\ & f(\lambda_k) & \ddots & \vdots \\ & & \ddots & f^{(1)}(\lambda_k) \\ & & & f(\lambda_k) \end{bmatrix}, \tag{5}$$

with $\boldsymbol{J}_k$ being the kth Jordan block, $f^{(n)}$ denotes the n-th derivative of f and $\boldsymbol{A} = \boldsymbol{Z}\boldsymbol{J}\boldsymbol{Z}^{-1}$ being the Jordan canonical form of $\boldsymbol{A}$. Here, s and n_k denote the number of Jordan blocks and dimension of the kth block, respectively.

Now, let us introduce the following definition.

Definition 1. (minimum polynomial) *The minimal polynomial of $\boldsymbol{A} \in \mathbb{C}^{n\times n}$ is the unique monic polynomial ψ of lowest degree such that $\psi(\boldsymbol{A}) = 0$.*

By considering the Jordan canonical form of $\boldsymbol{A}$, it can be shown that

$$\psi(t) = \prod_{i=1}^{s}(t-\lambda_i)^{n_i}, \tag{6}$$

where $\lambda_1,\ldots,\lambda_s$ are the distinct eigenvalues of $\boldsymbol{A}$ and n_i is the dimension of the largest Jordon block in which λ_i appears. From (6), we can observe that ψ is identically zero on the spectrum of $\boldsymbol{A}$. In addition, the degree of ψ, $d = \deg(\psi) = \sum_{i=1}^{s} n_i$, is at most n. This is because the dimensions of the Jordan blocks of $\boldsymbol{A}$ have to add up to n [17].

An interesting characteristic of the minimal polynomial ψ, is that it divides any other polynomial p for which $p(\boldsymbol{A}) = 0$ [16]. In addition, as noted in [9], this polynomial allows for an *autoregressive* description of sequential signals shifts. That is, considering $\boldsymbol{x}(k+1) \triangleq \boldsymbol{A}\boldsymbol{x}(k)$, where $\boldsymbol{x}(k) = [x_1(k)\ \cdots\ x_n(k)]^T$, we have that $\boldsymbol{x}(d) = -\sum_{k=0}^{d-1}\alpha_k\boldsymbol{x}(k)$ for some coefficients $\{\alpha_k\}_{k=0}^{d-1}$ defined by ψ. Using this relation, the authors in [9] showed that a node only requires to know its own d previous values to compute further linear iterations. However, the minimal polynomial of a matrix not only provides an autoregressive property but also *defines* a unique polynomial associated with a given matrix function [16]. This fact is provided in the following definition.

Definition 2. (matrix function via Hermitian interpolation) *Let f be a function defined over the spectrum of $\boldsymbol{A} \in \mathbb{C}^{n\times n}$. Further, consider ψ as the minimal polynomial of $\boldsymbol{A}$. Then, there exists a unique polynomial p of degree less than* $\deg(\psi)$ *such that*

$$p^{(j)}(\lambda_i) = f^{(j)}(\lambda_i), \quad j\in\{0,\ldots,n_i-1\}, \quad i\in\{1,\ldots,s\}, \tag{7}$$

and this polynomial is known as the Hermite interpolating polynomial.

Using these definitions, it is clear that any function f, defined in the spectrum of $\boldsymbol{A}$ [cf. (4)], has a unique polynomial of order $\deg(\psi)$ equivalent on the spectrum of $\boldsymbol{A}$. In the following section, we show how this polynomial can be directly used to generalize previous results in finite-time consensus using the general framework of matrix function theory.

3. FINITE-TIME CONSENSUS AND THE HERMITE POLYNOMIAL

Using the formalism introduced in the previous section, we can relate results from finite-time consensus with matrix functions theory.

For a matrix $\boldsymbol{A}$ with distinct eigenvalues, i.e., $n_i = 1$, $s = n$, the Hermite interpolating polynomial p [cf. Def. 2] is given in its Lagrange form by [16]

$$p(t) = \sum_{i=1}^{n} f(\lambda_i)\ell_i(t), \quad \ell_i(t) = \prod_{j=1, j\neq i}^{n}\Big(\frac{t-\lambda_j}{\lambda_i-\lambda_j}\Big). \tag{8}$$

From the Lagrange form of the polynomial (8), we can devise a way to achieve consensus in finite steps for certain networks. That is, consider the Laplacian matrix $\boldsymbol{L}$ and the function

$$f(\lambda) = \begin{cases} 1, & \lambda = 0 \\ 0, & \text{otherwise} \end{cases}, \tag{9}$$

where zero is the eigenvalue associate to the constant eigenvector $\boldsymbol{c}$ of $\boldsymbol{L}$ whose entries are all one[1].

Using this function and the spectrum of $\boldsymbol{L}$, (8) simplifies as

$$p(t) = \prod_{j=2}^{n}\Big(1-\frac{1}{\lambda_j}t\Big), \tag{10}$$

where we assigned an ordering to the spectrum of $\boldsymbol{L}$ in which $\lambda_1 = 0$. By substituting t with $\boldsymbol{L}$, we obtain the matrix polynomial

$$p(\boldsymbol{L}) = \prod_{j=2}^{n}\Big(\boldsymbol{I}-\frac{1}{\lambda_j}\boldsymbol{L}\Big), \tag{11}$$

where $\boldsymbol{I}$ is the $n\times n$ identity matrix. Notice that this expression is the polynomial description of the result provided in [10] but obtained by the natural framework of matrix functions. Here, there is no need to further prove that (11) achieves consensus in finite time as $p(\boldsymbol{L})$ is identically (9) by definition (which in turn defines the consensus operation when it is defined over the spectrum of $\boldsymbol{L}$). From (11) it is also seen that only $n-1$ diffusions are required, i.e., matrix multiplications.

In a similar way, we can extend these so-called graph filters to general matrices for which an eigenvalue decomposition is not defined or their eigenvalues have a multiplicity greater than one. That is, for a given function f we can obtain its Hermite interpolating polynomial [cf. Def. 2] of order $\deg(\psi)$ as

$$\begin{aligned} q(t) =& f[\lambda_1] + f[\lambda_1,\lambda_2](t-\lambda_1) + f[\lambda_1,\lambda_2,\lambda_3](t-\lambda_1)(t-\lambda_2) \\ &+ \cdots + f[\lambda_1,\ldots,\lambda_s](t-\lambda_1)(t-\lambda_2)\cdots(t-\lambda_{s-1}), \end{aligned} \tag{12}$$

where $f[\cdot]$ are the Newton's divided differences of the function f that are defined recursively by

$$f[x_k] = f(x_k) \tag{13}$$

$$f[x_k, x_{k+1}] = \begin{cases} \frac{f(x_{x+1})-f(x_k)}{x_{k+1}-x_k}, & x_k \neq x_{k+1} \\ f^{(1)}(x_{k+1}), & x_k = x_{k+1}, \end{cases} \tag{14}$$

$$f[x_0, x_1, \ldots, x_{k+1}] = \begin{cases} \frac{f[x_1,\ldots,x_k]-f[x_0,x_1,\ldots,x_k]}{x_{k+1}-x_0}, & x_0 \neq x_{k+1}, \\ \frac{f^{(k+1)}(x_{k+1})}{(k+1)!}, & x_0 = x_{k+1}. \end{cases} \tag{15}$$

[1] Here, we assume that zero is a simple eigenvalue, i.e., the multiplicity is equal to one, which implies a single connected component in the network.

The construction in (12) provides a way to compute the weights that a given implementation, in terms of graph filtering, needs to apply for computing the consensus operation. This approach exhibits connections to the construction proposed in [9] where the coefficients of the minimal polynomial of a given network have to be computed to implement a protocol using local linear aggregations for achieving consensus in finite time.

Notice that in the definition of the Newton's divided difference, the derivatives of f are required. This requirement can be waived in case of finite-dimensional matrices as it is possible to find a sufficiently differentiable function $\tilde{f}$ such that the interpolation condition [cf. (7)] is satisfied.

At this point, we have shown the natural connection between Hermitian interpolation, graph filters, and finite-time consensus. It should be clear now that given a fixed matrix $\boldsymbol{A}$, *nothing* can be faster than the application of the minimal polynomial of such a matrix if we are restricted to operations that can be defined as matrix functions. However, this approach might be too restrictive for general cases. In the following, we leverage certain structural properties to improve the performance achievable when a *known support*, i.e., nonzero positions, is given.

4. MINIMIZATION OF THE ORDER OF MINIMUM POLYNOMIAL

As discussed before, the order of the minimum polynomial defines the lowest order for any matrix function. Therefore, it is natural to raise the question: *is it possible to design the minimum polynomial for a given network?*

In this context, we might consider the following problem: given a fixed support, i.e., connections among nodes in the graph, can we design the *weights* of the network such that the resulting matrix has the minimum polynomial with the lowest order? Obviously, this problem has a trivial solution: the all-zero matrix. Despite this, in the following, we show that a *solution generator*, i.e., a descriptor for all solutions, can be obtained for this problem. Here, a notice of warning. As we are solving for a solution generator, we can expect that this approach might not be tractable for several instances.

First, let us consider a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ with entries $a_{i,j}, \forall\, i,j \in \{1,\dots,n\}$. Its zero entries are defined through the set $\mathcal{A}_o$, i.e., $\mathcal{A}_o = \{a_{i,j} \mid a_{i,j} = 0\}$. Further, let

$$\varphi(\lambda, \boldsymbol{A}) = \det(\boldsymbol{A} - \lambda \boldsymbol{I}), \tag{16}$$

be the characteristic polynomial of $\boldsymbol{A}$. Here, we have specified the dependency of the polynomial in terms of the entries of $\boldsymbol{A}$ by adding it as an argument of the function.

Now, consider the following true statement: $\boldsymbol{A}$ *has duplicated eigenvalues* $\iff$ $\varphi(\lambda, \boldsymbol{A})$ *has repeated roots.* This statement implies that the $\gcd(\varphi, \varphi')$ is the polynomial defined by the repeated roots of φ. Here, $\gcd(\cdot,\cdot)$ is the greatest common divisor of its arguments, and $\varphi'(\lambda, \boldsymbol{A})$ denotes the derivative, with respect to λ, of $\varphi(\lambda, \boldsymbol{A})$.

Therefore, we can transform the problem at hand into an *assignment problem*. That is, we need to find entries $\{a_{i,j}\}$ such that $\gcd(\varphi, \varphi')$ has the largest possible degree. At first glance, the only thing that has happened is that now instead of looking for some low degree polynomial, we are searching for a high degree polynomial. However, under this formulation, we can exploit the properties of *subresultants* [18].

Definition 3. (*i*th subresultant polynomial) *The ith subresultant polynomial, $S_i(p,q)$, of two polynomials p and q is a polynomial of degree at most i whose coefficients are polynomial functions of the coefficients of p and q.*

Algorithm 1: Algebraic Polynomial Order Maximization

Input: φ
Result: d^* : maximum order of $\gcd(\varphi, \varphi')$,
$\{a_{ij}\}$: assignment
Init: compute φ', $\gcd(\varphi, \varphi')$, $d = 0$;
while *consistent* **do**
  $d \leftarrow d + 1$;
  compute $S_d(\varphi, \varphi')$;
  $\{a_{ij}\} \leftarrow$ *solve system* $\{S_i(\varphi, \varphi') = 0\}_{i=1}^{d}$;
  if *inconsistent* **then**
    return $d^* \leftarrow d - 1$ break;
  end
end

The following property holds for subresultants.

Proposition 1. *[18] The* gcd *of p and q has a degree d if and only if the system*

$$S_0(p,q) = \cdots = S_{d-1}(p,q) = 0, \tag{17}$$

has a nontrivial solution.

This result implies that $S_d(p,q)$ is a $\gcd(p,q)$, hence the order of the $\gcd(p,q)$ is d which relates to the number of repeated eigenvalues. Using the necessary and sufficient condition provided in Proposition 1, we can devise an algebraic algorithm for finding a solution generator to maximize d, and therefore increase the number of repeated roots of the characteristic polynomial.

The algebraic algorithm is conceptually simple as it only requires to solve a series of systems of polynomial equations. Intuitively, we only need to solve the system in (17) for increasing values of d until it becomes inconsistent. This way, we find the maximum value of d for which the system is consistent. The procedure is summarized in Algorithm 1.

To solve such systems of polynomials we first compute a Gröbner basis [19] and check if the system is inconsistent, zero-dimensional or positive-dimensional [19]. Then, we solve the system by deducing the lexicographical Gröbner basis by the FGLM algorithm [20] and applying the Lextriangular algorithm [21]. For each consistent system, this provides an assignment $\{a_{ij}\}$ for the entries of the matrix, providing a solution for $\boldsymbol{A}$.

Although we devise an algorithm to approach the general problem, we remark that, under the current state-of-the-art algebraic geometry computational methods, Algorithm 1 has certain limitations. First, as we will most likely work with rational coefficients, the solution generator may involve large integers which hardens the computations. Second, as we are not only interested in the *consistency* of the problem, we need to deduce the numeric values for a particular solution. This requires a solution to univariate polynomials whose coefficients are approximated, i.e., finite-arithmetic precision, hence it is a highly unstable problem. Despite that research has been done to tackle such problems (see [22, 23]), this approach still requires a large computational effort. Hence, for many problem instances this approach might not be computationally tractable.

Although the introduced constructive method presents computational challenges, in the following, we focus on a particular instance of the problem which still is of interest. Here, besides knowledge of the support of the network more information of its structure is

employed to obtain the network with the largest possible number of repeated eigenvalues.

5. A PARTICULAR CASE: KNOWN GRAPH EIGENBASIS

In the previous section, we have shown that it is possible to increase the number of repeated eigenvalues for a fixed-support matrix using algebraic techniques. However, most of the times, this approach might be intractable. As a result, we require to impose restrictions to make this problem tractable.

Here, we follow the framework of [15,24] where instead of considering the family of matrices with a fixed support, we consider the family of matrices that have a *fixed support and are diagonalizable with the same set of eigenvectors*. That is, we focus on matrices that belong to the set

$$\mathcal{M}_{\boldsymbol{U}}^{\mathcal{A}_0} = \{\boldsymbol{A} : \boldsymbol{U}^{-1}\boldsymbol{A}\boldsymbol{U} = \text{diag}(\boldsymbol{\omega}),\ \text{zero}(\boldsymbol{A}) = \mathcal{A}_0\}, \tag{18}$$

where $\boldsymbol{U}$ is an $n \times n$-eigenbasis, $\text{diag}(\cdot)$ constructs a diagonal matrix out of it vector argument, and $\text{zero}(\cdot)$ denotes the indexes of the zero entries of its argument.

For matrices belonging to the set (18), the problem of minimizing the degree of the minimum polynomial can be further simplified. Instead of the general problem, we can now ask the following question: *what is the matrix with support defined by $\mathcal{A}_0$, and with eigenbasis $\boldsymbol{U}$, that has the largest number of repeated eigenvalues?*

As noted in [15], the matrices belonging to (18) can be parametrized using the nullspace of a given matrix. More specifically, the elements of (18) have eigenvalues given by $\boldsymbol{\omega} = \boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}\boldsymbol{\alpha}$, where $\boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}$ is a basis for the nullspace of $\boldsymbol{\Phi}_{\mathcal{A}_0}(\boldsymbol{U}^{-T} * \boldsymbol{U})$, where $\boldsymbol{\Phi}_{\mathcal{A}_0}$ is the $|\mathcal{A}_0| \times n^2$-selection matrix whose rows are the rows of the $n^2 \times n^2$ identity matrix indexed by the set $\mathcal{A}_0$. Here, $|\cdot|$ denotes the cardinality of a set. The dimension of this nullspace is fixed given the pair $(\mathcal{A}_0, \boldsymbol{U})$, i.e., $\dim(\boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}) = r$.

The following proposition provides a result involving the dimension of this nullspace and the number of repeated eigenvalues.

Lemma 1. *Let $\boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}$ be an $n \times r$-basis for the nullspace of the matrix*

$$\boldsymbol{\Phi}_{\mathcal{A}_0}(\boldsymbol{U}^{-T} * \boldsymbol{U})$$

for a given eigenbasis $\boldsymbol{U}$ and support defined by $\mathcal{A}_0$. Then there exists a matrix with eigenbasis $\boldsymbol{U}$ and support defined by $\mathcal{A}_0$ with at least $r-1$ repeated eigenvalues and it is not the all-zero matrix.

Proof. The proof follows from the rank condition of $\boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}$. By selecting r linearly independent rows, we can build a nontrivial $r \times r$ linear system which has a unique solution. For instance, for the consensus problem, we can consider the system $\boldsymbol{B}\boldsymbol{z} = \boldsymbol{b} = [1, 0, 0, \ldots, 0]^T$, where $\boldsymbol{B}$ denotes a submatrix of $\boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}$ with r rows and $\boldsymbol{b}$ is a target partial spectrum. As $\boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}$ is full column rank, there exists a $\boldsymbol{B}$ such that $\text{rank}(\boldsymbol{B}) = r$. Therefore, the above system has a (unique) exact solution $\boldsymbol{z}^*$ as desired. As the first entry of $\boldsymbol{b}$ is nonzero, the resulting matrix with partial spectrum $\boldsymbol{b}$ is not all-zero. □

Lemma 1 leads to the following interpretation: *There exists a graph filter, i.e., a polynomial on the matrix representation of the graph, which can be implemented in a single step and nulls at least $r-1$ graph modes.* This fact arises naturally as it is clear that all matrices in (18) are *polynomials* of the *graph shift operator*, i.e., a matrix representation of the graph from which the eigenbasis has been extracted, and they all share the support of the shift. Therefore, they are a particular family of polynomials that can be implemented through point-wise weighting of the shift operator. Using the result of Lemma 1 and its connection with graph filters, we can state the following result on finite-time consensus.

Proposition 2. *Let $\mathcal{G}$ be a connected (network) graph, and $\boldsymbol{W}$ be a $n \times n$ symmetric matrix satisfying*

$$w_{nn} = \sum_{m \in \mathcal{N}_n} w_{nm} \quad \text{and} \quad w_{nm} \leq 0 \quad n \neq m, \tag{19}$$

where $\mathcal{N}_n$ is the set of (vertices) nodes that has an edge with the nth node, and having distinct eigenvalues. Furthermore, let r be the rank of $\boldsymbol{B}_{\boldsymbol{U}}^{\mathcal{A}_0}$, where $\boldsymbol{U}$ and $\mathcal{A}_0$ are the eigenbasis and the set of zero entries of $\boldsymbol{W}$, respectively.

Then, the minimum number of communication rounds, k^*, required to compute the average in a distributed manner is upper bounded as

$$k^* \leq \begin{cases} n - r + 1 & r \geq 2 \\ n - 1 & \textit{otherwise} \end{cases} \tag{20}$$

Proof. Under the given conditions, the matrix $\boldsymbol{W}$ can be seen as a Laplacian matrix for the graph. Further, since the graph is connected, the multiplicity of its zero eigenvalue is one. In addition, the all-one vector is the related eigenvector to this eigenvalue. Therefore, consensus can be computed using this communication matrix. Now, for the case $r < 2$, we can directly apply the polynomial (10) whose implementation only requires $n-1$ communication rounds. This result is equivalent to the result provided by [10]. For $r \geq 2$, from Lemma 1, there exists a graph filter that nulls at least $r-1$ eigenvalues in a single communication round, i.e., there exists a matrix with the same support as the connections of the network which zeroes several eigenvalues in a single operation. As a result, after applying this graph filter, we can apply a truncated version of the polynomial (10), i.e., only considering the eigenvalues that have not been nulled, to null the remaining $N - (r+1)$ eigenvalues. Hence, the result. □

Proposition 2 generalizes the state-of-the-art in finite-time consensus through linear operations on the matrix representation of the graph as in the case of a graph with $r > 2$, we obtain $k^* < \deg(\psi)$. This result provides guarantees for faster convergence in certain networks although the original graph from which the eigenbasis has been taken has no repeated eigenvalues. Further, it provides a venue to explore the capabilities of weight-varying networks to improve convergence to consensus. That is, finding network weights, for each communication round, such that more than one eigendirection gets nulled, as already demonstrated in [14, 15].

6. CONCLUSION

Using matrix function theory, we generalized all state-of-the-art methods for finite-time consensus through linear operations of the matrix representation of a network. This framework provides a unified strategy for designing consensus algorithms for arbitrary networks. We showed that it is possible to minimize the order of the minimum polynomial of a network with known support through algebraic geometry techniques. As this approach is generally not tractable, we showed that, under certain conditions, we can achieve faster consensus convergence than stated in known bounds. We achieved this by restricting our discussion to a particular family of matrices who do not only share the same support but also the same eigenbasis.

7. REFERENCES

[1] Morris H DeGroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.

[2] John Tsitsiklis, Dimitri Bertsekas, and Michael Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," IEEE *Trans. Autom. Control*, vol. 31, no. 9, pp. 803–812, 1986.

[3] Ali Jadbabaie, Jie Lin, and A Stephen Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on automatic control*, vol. 48, no. 6, pp. 988–1001, 2003.

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

[5] Lin Xiao and Stephen Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.

[6] Alain Y Kibangou, "Finite-time average consensus based protocol for distributed estimation over awgn channels," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*. IEEE, 2011, pp. 5595–5600.

[7] Shreyas Sundaram and Christoforos N Hadjicostis, "Distributed function calculation and consensus using linear iterative strategies," *IEEE journal on selected areas in communications*, vol. 26, no. 4, 2008.

[8] Sérgio Pequito, Stephen Kruzick, Soummya Kar, José MF Moura, and A Pedro Aguiar, "Optimal design of distributed sensor networks for field reconstruction," in *21st European Signal Processing Conference (EUSIPCO 2013)*. IEEE, 2013, pp. 1–5.

[9] Shreyas Sundaram and Christoforos N Hadjicostis, "Finite-time distributed consensus in graphs with time-invariant topologies," in *American Control Conference, 2007. ACC'07*. IEEE, 2007, pp. 711–716.

[10] Aliaksei Sandryhaila, Soummya Kar, and José MF Moura, "Finite-time distributed consensus through graph filters," in *Proc.* IEEE *Int. Conf. Acoust., Speech Signal Process. (ICASSP)*. IEEE, 2014, pp. 1080–1084.

[11] Aliaksei Sandryhaila and José MF Moura, "Discrete signal processing on graphs," IEEE *Trans. Signal Process*, vol. 61, no. 7, pp. 1644–1656, 2013.

[12] Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[13] Mario Coutino, Elvin Isufi, and Geert Leus, "Distributed Edge-Variant Graph Filters," in *Int. Workshop on Comp. Adv. in Multi-Sensor Adaptive Proc. (CAMSAP)*. IEEE, 2017.

[14] Santiago Segarra, Antonio Marques, and Alejandro Ribeiro, "Optimal graph-filter design and applications to distributed linear network operators," IEEE *Trans. Signal Process*, 2017.

[15] Mario Coutino, Elvin Isufi, and Geert Leus, "Advances in distributed graph filtering," *arXiv preprint arXiv:1808.03004*, 2018.

[16] Nicholas J Higham, *Functions of matrices: theory and computation*, vol. 104, Siam, 2008.

[17] Roger A Horn and Charles R Johnson, *Matrix analysis*, Cambridge university press, 2012.

[18] Saugata Basu, Richard Pollack, and Marie-Françoise Coste-Roy, *Algorithms in real algebraic geometry*, vol. 10, Springer Science & Business Media, 2007.

[19] Daniel Lazard, "Gröbner bases, gaussian elimination and resolution of systems of algebraic equations," in *European Conference on Computer Algebra*. Springer, 1983, pp. 146–156.

[20] Jean-Charles Faugere, Patrizia Gianni, Daniel Lazard, and Teo Mora, "Efficient computation of zero-dimensional gröbner bases by change of ordering," *Journal of Symbolic Computation*, vol. 16, no. 4, pp. 329–344, 1993.

[21] Daniel Lazard, "Solving zero-dimensional algebraic systems," *Journal of symbolic computation*, vol. 13, no. 2, pp. 117–131, 1992.

[22] Xavier Dahan and Éric Schost, "Sharp estimates for triangular sets," in *Proceedings of the 2004 international symposium on Symbolic and algebraic computation*. ACM, 2004, pp. 103–110.

[23] Xavier Dahan, Marc Moreno Maza, Eric Schost, Wenyuan Wu, and Yuzhen Xie, "Lifting techniques for triangular decompositions," in *Proceedings of the 2005 international symposium on Symbolic and algebraic computation*. ACM, 2005, pp. 108–115.

[24] Mario Coutino, Sundeep Chepuri, and Geert Leus, "Sparsest network support estimation: A submodular approach," in *2018 IEEE Data Science Workshop*, 2018.