

Fast Distributed Subspace Projection via Graph Filters

Thilina Weerasinghe, Daniel Romero, César Asensio-Marco, and Baltasar Bekerull-Lozano

Intelligent Signal Processing and Wireless Networks (WISENET) Lab

Department of Information and Communication Technology and CIEM

University of Agder, 4879, Grimstad, Norway

Email: {daniel.romero, thilina.weerasinghe, cesar.asensio, baltasar.bekerull}@uia.no

Abstract—A significant number of linear inference problems in wireless sensor networks can be solved by projecting the observed signal onto a given subspace. Decentralized approaches avoid the need for performing such an operation at a central processor, thus, reducing the congestion and increasing the robustness of the communication network. Unfortunately, existing decentralized approaches either confine themselves to a reduced family of subspace projection tasks or need an infinite number of iterations to obtain the exact projection. To remedy these limitations, this paper develops a framework for computing a wide class of subspace projections in a decentralized fashion by relying on the notion of graph filtering. To this end, a methodology to obtain the shift matrix and the corresponding filter coefficients that provide exact subspace projection in a nearly minimal number of iterations is proposed. Numerical experiments corroborate the merits of the proposed approach.

I. INTRODUCTION

Wireless sensor networks (WSNs) perform inference tasks in applications demanding distributed monitoring and operation. Many of these tasks, such as least squares estimation, denoising, weighted consensus, and distributed detection [1], [2] to name a few, can be cast as projecting the observed signal onto a subspace known to contain the true signal. Although subspace projection can be performed in a centralized manner if all nodes send their measurements to a fusion center, such scheme is not robust against nodes failures and may provoke congestion. These limitations motivate the use of novel methodologies to implement the subspace projection in a distributed manner.

Distributed subspace projection through information exchanges among neighbouring nodes has been addressed in [3]. At every iteration of the proposed method, each node linearly combines its previous iterate with the ones of its neighbors. The weights of this linear combination are adjusted to achieve a fast asymptotic convergence. In a complementary way, the authors in [1] propose a method to obtain the aforementioned weights in a distributed manner. Unfortunately, these approaches only give rise to asymptotic convergence of the iterative method to the subspace projection solution. Therefore, a significant number of transmissions repeated over a long period of time are needed to reduce the error below a given bound.

In addition, these approaches can only be applied to a rather reduced set of topologies, as shown in [4]. These limitations have been alleviated for consensus, which is a special case of subspace projection, in the literature of graph signal processing [5] through graph filters [6], [7], [8]. Graph filters generalize classical time-invariant filters to accommodate signals defined on the vertices of a graph. Remarkably, these approaches for distributed consensus converge in a finite number of iterations. More general scenarios of subspace projection have been addressed in [7] and [9], but the proposed schemes require knowledge of the so-called *shift matrix*. Unfortunately, a valid shift matrix to perform a given projection task is seldom known, which limits the applicability of these approaches. To the best of our knowledge, there is no complete methodology in the literature to perform general subspace projection in a finite number of iterations.

This paper develops a method to obtain not only the graph filter coefficients that enable distributed subspace projection through graph filters in a finite number of iterations. In addition, the order of the resulting filters, which equals the number of communications needed between each pair of connected nodes, is minimized.

The remainder of the paper is structured as follows. Sec. II, reviews some background related to subspace projection and graph filters before formulating the problem. Sec. III describes the proposed algorithm. Finally, Sec. IV presents some numerical results and Sec.V concludes the paper.

Notation: Vectors (respectively matrices) are denoted by bold lowercase (uppercase) letters. $\mathbf{1}$ represents the vector of all ones and $\mathbf{0}$ the vector of all zeros. The spectral radius of a matrix \mathbf{A} is $\rho(\mathbf{A}) \triangleq \max\{|\lambda_1|, \dots, |\lambda_n|\}$ where $\lambda_1, \dots, \lambda_n$ are the eigenvalues of \mathbf{A} . The 2-norm of matrix \mathbf{A} is $\|\mathbf{A}\|_2 = \sqrt{\lambda_{max}(\mathbf{A}^* \mathbf{A})} = \sigma_{max}(\mathbf{A})$, where λ_{max} (respectively σ_{max}) denote the largest eigenvalue (singular value). Finally, \otimes denotes Kronecker product, $\|\mathbf{A}\|_*$ the nuclear norm of a matrix \mathbf{A} , $\text{cols}(\mathbf{A})$ the columns of matrix \mathbf{A} , $\mathcal{R}(\mathbf{A})$ the span of the columns of \mathbf{A} , and $\text{evals}(\mathbf{A})$ the set of eigenvalues of \mathbf{A} .

II. PROBLEM FORMULATION

After reviewing distributed subspace projection and graph filters, this section formulates the problem.

A. Distributed Subspace Projection

Consider a network of N sensor nodes represented by a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{v_1, \dots, v_N\}$ the set of vertices and

The work in this paper was supported by the PETROMAKS Smart-Rig grant 244205/E30, the SFI Offshore Mechatronics grant 237896/O30, and the TOPPFORSK WISECART grant 250910/F20.

$\mathcal{E} \subset V \times V$ the set of edges. Two vertices are connected by an edge if and only if the associated nodes can communicate directly. Let \mathbf{A} denote the adjacency matrix of \mathcal{G} , where $(\mathbf{A})_{n,n'} = 1$ if $(v_n, v_{n'}) \in \mathcal{E}$ and $(\mathbf{A})_{n,n'} = 0$ otherwise. In this paper, it is assumed that node v_n can communicate with node $v_{n'}$ if node $v_{n'}$ can communicate with node v_n , that is, \mathcal{G} is undirected. Moreover, it will be assumed that \mathcal{G} contains all self-loops, that is $(v_n, v_n) \in \mathcal{E} \forall n$, since it is implicit that any node can communicate with itself. Given the observation vector $\mathbf{z} = [z_1, \dots, z_N]^T$, where z_n denotes the observation of node $v_n \in \mathcal{V}$, the goal is to estimate the signal vector $\boldsymbol{\xi} \in \mathbb{R}^N$ modeling the phenomenon of interest (e.g. temperature), which is related to \mathbf{z} via:

$$\mathbf{z} = \boldsymbol{\xi} + \mathbf{v}, \quad (1)$$

where $\mathbf{v} \in \mathbb{R}^N$ stands for additive noise.

In subspace projection tasks, $\boldsymbol{\xi}$ is known to lie in some subspace of dimension $r < N$. Let $\mathbf{U}_{\parallel} \in \mathbb{R}^{N \times r}$ be a matrix whose columns span a subspace. Hence, vector $\boldsymbol{\xi}$ can be expressed as $\boldsymbol{\xi} = \mathbf{U}_{\parallel} \boldsymbol{\alpha}$ for some $\boldsymbol{\alpha} \in \mathbb{R}^r$. Without loss of generality, the columns of \mathbf{U}_{\parallel} are assumed orthonormal. The orthogonal projection of \mathbf{z} onto the subspace spanned by the columns of \mathbf{U}_{\parallel} equals the least-squares estimate of $\boldsymbol{\xi}$ and will be denoted by $\hat{\boldsymbol{\xi}}$. It follows that:

$$\hat{\boldsymbol{\xi}} \triangleq [\hat{\xi}_1, \dots, \hat{\xi}_N]^T = \mathbf{U}_{\parallel} \mathbf{U}_{\parallel}^T \mathbf{z} \triangleq \mathbf{P} \mathbf{z} \quad (2)$$

where $\mathbf{P} \in \mathbb{R}^{N \times N}$ is the projection matrix. The goal of this operation is to find $\boldsymbol{\xi}$, given \mathbf{z} and \mathbf{U}_{\parallel} . In practice, the dimension r of the useful signal subspace is typically much smaller than the dimension N of the observation space. Hence, a strong noise reduction may arise from the projection in (2) [3]. Besides least squares estimation, another particular example of subspace projection is the average consensus, where the projection matrix is $\mathbf{P} = \mathbf{1}\mathbf{1}^T/N$.

A distributed scheme for subspace projection is proposed in [3], where the iterates $\mathbf{z}[k+1] = \mathbf{W}\mathbf{z}[k]$ are iteratively computed for $k = 0, 1, \dots$ with initialization $\mathbf{z}[0] = \mathbf{z}$. Matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$ satisfying $(\mathbf{W})_{n,n'} = 0$ if $(v_n, v_{n'}) \notin \mathcal{E}$, is sought to satisfy $\lim_{k \rightarrow \infty} \mathbf{z}[k] = \lim_{k \rightarrow \infty} \mathbf{W}^k \mathbf{z} = \mathbf{P}\mathbf{z}$, $\forall \mathbf{z} \in \mathbb{R}^N$ i.e. $\lim_{k \rightarrow \infty} \mathbf{W}^k = \mathbf{P}$. The number of local exchanges for convergence is high since this criterion targets asymptotic convergence and, moreover, the set of feasible topologies is limited. The present paper considers approaches that converge to the exact projection in a finite number of iterations and for a larger set of feasible topologies.

B. Distributed subspace projection via graph filters

To introduce the notion of graph filter, it is necessary to define the so-called graph shift operator $\mathbf{z} \mapsto \mathbf{S}\mathbf{z}$, where the symmetric matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$ satisfies $(\mathbf{S})_{n,n'} = 0$ if $(v_n, v_{n'}) \notin \mathcal{E}$, and is referred to as the shift matrix [7]. Examples of shift matrices include \mathbf{A} and the combinatorial Laplacian $\text{diag}[\mathbf{A}\mathbf{1}] - \mathbf{A}$.

This shift operator can be evaluated distributedly. To see this observe that the n -th entry of $\mathbf{y} \triangleq \mathbf{S}\mathbf{z}$ can be expressed as

$$y_n = \sum_{n'=1}^N (\mathbf{S})_{n,n'} z_{n'} = \sum_{n':(v_n, v_{n'}) \in \mathcal{E}} (\mathbf{S})_{n,n'} z_{n'} \quad (3)$$

where the second equality follows from the definition of shift matrix. Thus, to compute y_n , node v_n only needs the entries of \mathbf{z} corresponding to its neighbors, which can be obtained through local information exchanges. Therefore \mathbf{y} can be computed in a decentralized fashion. Observe that the operator $\mathbf{z} \mapsto \mathbf{S}^l \mathbf{z}$ can also be evaluated in a distributed manner by iteratively applying the shift operator l times: if $\mathbf{z}[0] = \mathbf{z}$ and $\mathbf{z}[k] = \mathbf{S}\mathbf{z}[k-1], k = 1, \dots, l$, it follows that $\mathbf{z}[l] = \mathbf{S}^l \mathbf{z}$. A graph filter takes this idea one step further by introducing also linear combinations of $\{\mathbf{z}[k]\}_{k=0}^l$. Specifically, an order- L graph filter is a polynomial of degree $L-1$ of the graph shift operator \mathbf{S} of the form:

$$\mathbf{H} := \sum_{l=0}^{L-1} c_l \mathbf{S}^l, \quad (4)$$

where $\{c_l\}_{l=0}^{L-1}$ are the filter coefficients.

This paper develops an efficient methodology to implement the subspace projection operation in a distributed fashion using graph filters, that is, finding $\{c_l\}_{l=0}^{L-1}$ and \mathbf{S} such that $\mathbf{H} = \mathbf{P}$. By introducing memory, we show that graph filters provide faster distributed subspace projection than the scheme in [3]. In [6], finite-time distributed consensus was achieved using graph filters. Also in [7], a general framework to implement arbitrary linear operators using graph filters is developed and applied to consensus and network coding. However, both previous approaches rely on the knowledge of a suitable shift matrix, but a methodology to obtain a valid shift matrix for general subspace projection remains an open problem and is the focus of this paper. Among feasible shift matrices, the one leading to the smallest order L will be selected. The problem can thus be formulated as: given \mathbf{P} and \mathcal{E} , find $\mathbf{S} \in \mathbb{R}^{N \times N}$ and $\{c_l\}_{l=0}^{L-1}$ such that $(\mathbf{S})_{n,n'} = 0$ if $(v_n, v_{n'}) \notin \mathcal{E}$, $n, n' = 1, \dots, N$ and $\mathbf{P} = \sum_{l=0}^{L-1} c_l \mathbf{S}^l$ with L as small as possible.

III. SHIFT MATRICES FOR FASTEST DISTRIBUTED SUBSPACE PROJECTION

This section develops an approach to obtain the shift matrix and the corresponding filter coefficients that provide exact subspace projection in a minimal number of iterations. For this purpose, we first characterize the set of feasible shift matrices to be used during the filtering process. Then, an approach to compute the corresponding filter coefficients is presented. Finally, an optimization methodology is proposed to minimize the order of the filter (i.e. the number of iterations needed to perform the projection via the graph filtering process).

A. Computation of the filter coefficients

The set of feasible shift matrices is given by:

$$\mathcal{S} := \left\{ \mathbf{S} \in \mathbb{R}^{N \times N} \text{ such that } (\mathbf{S})_{n,n'} = 0 \text{ if } (v_n, v_{n'}) \notin \mathcal{E}, \right. \\ \left. \exists L, \mathbf{c} = [c_0, \dots, c_{L-1}]^T \text{ satisfying } \mathbf{U}_{\parallel} \mathbf{U}_{\parallel}^T = \sum_{l=0}^{L-1} c_l \mathbf{S}^l \right\} \quad (5)$$

The next result accomplishes the characterization of the shift matrices involved in the graph filtering process for subspace projection.

Proposition 1. Let $\mathbf{U}_{\parallel} \in \mathbb{R}^{N \times r}$ have orthonormal columns and let \mathcal{S} be given by (5). Then,

$$\begin{aligned} \mathcal{S} &= \left\{ \mathbf{S} : \mathbf{S} = \mathbf{U}_{\parallel} \mathbf{E} \mathbf{A}_{\parallel} \mathbf{E}^T \mathbf{U}_{\parallel}^T + \mathbf{V}_{\perp} \mathbf{A}_{\perp} \mathbf{V}_{\perp}^T \text{ for some} \right. \\ \mathbf{E} &\in \mathbb{R}^{r \times r}, \mathbf{A}_{\parallel} \in \mathbb{R}^{r \times r} \text{ diagonal}, \mathbf{A}_{\perp} \in \mathbb{R}^{(N-r) \times (N-r)} \text{ diagonal}, \\ \mathbf{V}_{\perp} &\in \mathbb{R}^{N \times (N-r)} \text{ satisfying } \mathbf{E} \mathbf{E}^T = \mathbf{I}, \mathbf{V}_{\perp}^T \mathbf{V}_{\perp} = \mathbf{I}, \mathbf{U}_{\parallel}^T \mathbf{V}_{\perp} = \mathbf{0}, \\ &(\mathbf{A}_{\parallel})_{n,n} \neq (\mathbf{A}_{\perp})_{n',n'}, \forall n \in \{1, \dots, r\}, \forall n' \in \{1, \dots, N-r\} \left. \right\} \\ &= \left\{ \mathbf{S} : \mathbf{S} = \mathbf{S}_{\parallel} + \mathbf{S}_{\perp} \text{ for some } \mathbf{S}_{\parallel} \text{ and } \mathbf{S}_{\perp} \text{ satisfying} \right. \\ &\quad \mathbf{S}_{\parallel} = \mathbf{S}_{\parallel}^T, \text{cols}(\mathbf{S}_{\parallel}) \in \mathcal{R}(\mathbf{U}_{\parallel}), \mathbf{S}_{\perp} = \mathbf{S}_{\perp}^T, \\ &\quad \text{cols}(\mathbf{S}_{\perp}) \perp \mathbf{U}_{\parallel}, \text{ either } \text{rank}(\mathbf{S}_{\parallel}) = r \text{ or } \text{rank}(\mathbf{S}_{\perp}) = N-r, \\ &\quad \left. \text{if } \lambda \in \text{evals}(\mathbf{S}_{\parallel}) \text{ and } \lambda \neq 0 \text{ then } \lambda \notin \text{evals}(\mathbf{S}_{\perp}); \right\} \quad (6) \end{aligned}$$

The proof is omitted here due to space restrictions. This proposition essentially establishes that a shift matrix \mathbf{S} satisfies (4) for some $\{c_l\}_{l=0}^{L-1}$ if it can be decomposed as the sum of two matrices \mathbf{S}_{\parallel} and \mathbf{S}_{\perp} which respectively span $\mathcal{R}(\mathbf{U}_{\parallel})$ and its orthogonal complement. Moreover, the non-null eigenvalues of \mathbf{S}_{\parallel} and \mathbf{S}_{\perp} must be different.

The next step is to determine $\{c_l\}_{l=0}^{L-1}$ if a valid \mathbf{S} is given. If $\mathbf{S} \in \mathcal{S}$, then Proposition 1 establishes that $\mathbf{P} = \mathbf{U}_{\parallel} \mathbf{U}_{\parallel}^T = \sum_{l=0}^{L-1} c_l \mathbf{S}^l$ amounts to:

$$\mathbf{U}_{\parallel} \mathbf{U}_{\parallel}^T = \mathbf{U}_{\parallel} \mathbf{E} \left[\sum_{l=0}^{L-1} c_l \mathbf{A}_{\parallel}^l \right] \mathbf{E}^T \mathbf{U}_{\parallel}^T + \mathbf{V}_{\perp} \left[\sum_{l=0}^{L-1} c_l \mathbf{A}_{\perp}^l \right] \mathbf{V}_{\perp}^T. \quad (7)$$

Multiplying both sides on the left by \mathbf{U}_{\parallel}^T and on the right by \mathbf{U}_{\parallel} , it follows that $\mathbf{I} = \mathbf{E} \left[\sum_{l=0}^{L-1} c_l \mathbf{A}_{\parallel}^l \right] \mathbf{E}^T$ or, equivalently, $\mathbf{I} = \sum_{l=0}^{L-1} c_l \mathbf{A}_{\parallel}^l$. On the other hand, multiplying (7) on the left by \mathbf{V}_{\perp}^T and on the right by \mathbf{V}_{\perp} , it follows that $\mathbf{0} = \sum_{l=0}^{L-1} c_l \mathbf{A}_{\perp}^l$. Therefore, one must have that

$$\begin{bmatrix} \mathbf{1}_r \\ \mathbf{0}_{N-r} \end{bmatrix} = \begin{bmatrix} 1 & \lambda_1 & \dots & \lambda_1^{L-1} \\ 1 & \lambda_2 & \dots & \lambda_2^{L-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_N & \dots & \lambda_N^{L-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{L-1} \end{bmatrix} \quad (8)$$

where $\lambda_1, \dots, \lambda_N$ are such that $\mathbf{A}_{\parallel} \triangleq \text{diag}\{\lambda_1, \dots, \lambda_r\}$ and $\mathbf{A}_{\perp} \triangleq \text{diag}\{\lambda_{r+1}, \dots, \lambda_N\}$. Equation (8) can be expressed in vector-matrix form as

$$\boldsymbol{\lambda}_p = \boldsymbol{\Psi} \mathbf{c} \quad (9)$$

Expression (9) provides a means to obtain the coefficients $\{c_l\}_{l=0}^{L-1}$ for a feasible \mathbf{S} .

B. Minimization of the order of the filter

The next goal is to minimize L . To this end, note that $\boldsymbol{\Psi}$ is Vandermonde and therefore its rank is $\min(L, N')$, where N' is the number of distinct eigenvalues in $\{\lambda_n\}_{n=1}^{L-1}$. Since $N' \leq N$, it follows that (9) is necessarily satisfied for some \mathbf{c} if $\mathbf{S} \in \mathcal{S}$ and $L = N$. For this reason, the approach adopted here will be to set $L = N$ and then find \mathbf{c} and \mathbf{S} such that the last entries of \mathbf{c} are zero.

Since $L = N$, matrix $\boldsymbol{\Psi}$ is invertible if $\lambda_n \neq \lambda_{n'}$ for all $n \neq n'$. In that case $\mathbf{c} = \boldsymbol{\Psi}^{-1} \boldsymbol{\lambda}_p$ and no entry of \mathbf{c} is necessarily zero. Therefore, the resulting filter $\sum_{l=0}^{N-1} c_l \mathbf{S}^l$ will generally be of order N . On the other hand, if there are replicated eigenvalues $\{\lambda_n\}_{n=1}^N$, one can exploit the resulting degrees of freedom to reduce the order of the filter. Suppose, for example, that $\lambda_1 = \lambda_2$ and that $\lambda_n \neq \lambda_{n'} \forall n, n' \in \{2, \dots, N\}, n \neq n'$. In that case, one can form matrix $\tilde{\boldsymbol{\Psi}}$ which is the result of replacing the first row of $\boldsymbol{\Psi}$ with $[0, 0, \dots, 0, 1]$, and also the vector $\tilde{\boldsymbol{\lambda}}_p$ by replacing the first entry of $\boldsymbol{\lambda}_p$ with a zero. Besides (9), the resulting $\mathbf{c} = \tilde{\boldsymbol{\Psi}}^{-1} \tilde{\boldsymbol{\lambda}}_p$ satisfies $c_{N-1} = 0$ and the corresponding filter is of order $N-1$. Similarly, if there are multiple repeated eigenvalues in $\{\lambda_n\}_{n=1}^N$, one can replace further rows of $\boldsymbol{\Psi}$ with vectors $[0, \dots, 0, 1, 0, \dots, 0]$ that ensure that the last entries of \mathbf{c} are zero. In general, the order of the fastest filter is N' . Hence, minimizing the number of distinct eigenvalues of \mathbf{S} minimizes the number of filter coefficients required and thereby the order of the filter. This observation motivates the search for shifts which minimize the number of distinct eigenvalues for fastest convergence. The rest of the section accomplishes this task.

From Sec. III, any feasible \mathbf{S} can be decomposed as $\mathbf{S} = \mathbf{S}_{\parallel} + \mathbf{S}_{\perp}$. The goal is to find a shift matrix \mathbf{S} whose eigenvalues are as equal as possible, yet the non-zero eigenvalues of \mathbf{S}_{\parallel} and \mathbf{S}_{\perp} must differ, cf. (6). A natural approach would be therefore to minimize the total number of different eigenvalues of \mathbf{S} subject to (i) \mathbf{S} must satisfy $(\mathbf{S})_{n,n'} = 0$ if $(v_n, v_{n'}) \notin \mathcal{E}$, and (ii) the non-zero eigenvalues of \mathbf{S}_{\parallel} and \mathbf{S}_{\perp} must differ. Unfortunately, this problem is non-convex due to the objective and due to (ii). The rest of this section describes a means to alleviate this difficulty. First, one can replace the aforementioned objective with a convex surrogate. To this end, note that if (ii) holds, the number of distinct non-zero eigenvalues of \mathbf{S} equals the number of distinct non-zero eigenvalues of \mathbf{S}_{\parallel} plus the number of distinct non-zero eigenvalues of \mathbf{S}_{\perp} . Focusing on \mathbf{S}_{\parallel} , note that the number of distinct non-zero eigenvalues of \mathbf{S}_{\parallel} equals the zero norm of the vector $[\lambda_1 - \lambda_2, \lambda_1 - \lambda_3, \dots, \lambda_1 - \lambda_r, \lambda_2 - \lambda_3, \dots, \lambda_{r-1} - \lambda_r]^T$. A convex surrogate of such a zero norm is the l_1 -norm [10]. However, one cannot use such an l_1 -norm directly as an objective since $\lambda_1, \dots, \lambda_r$ are generally non-convex functions of \mathbf{S} . To remedy this problem, one can adopt the objective $\|\mathbf{S}_{\parallel} \otimes \mathbf{I}_N - \mathbf{I}_N \otimes \mathbf{S}_{\parallel}\|_{\star}$. The nuclear norm equals the l_1 -norm of the singular values of its argument, and the latter comprise the absolute differences between pairs of eigenvalues of \mathbf{S}_{\parallel} . Specifically, it is easy to see that if $\mathbf{S} = \mathbf{U}_{\parallel} \mathbf{E} \mathbf{A}_{\parallel} \mathbf{E}^T \mathbf{U}_{\parallel}^T$ (cf. (7)), then \mathbf{A}_{\parallel} is the matrix of eigenvalues of \mathbf{S}_{\parallel} and $\|\mathbf{S}_{\parallel} \otimes \mathbf{I}_N - \mathbf{I}_N \otimes \mathbf{S}_{\parallel}\|_{\star} = \|\mathbf{A}_{\parallel} \otimes \mathbf{I}_N - \mathbf{I}_N \otimes \mathbf{A}_{\parallel}\|_{\star}$. Moreover, observe that

$$\mathbf{A}_{\parallel} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{A}_{\parallel} = \begin{bmatrix} \lambda_1 \mathbf{I} & & & \\ & \ddots & & \\ & & \lambda_r \mathbf{I} & \\ & & & \ddots \end{bmatrix} - \begin{bmatrix} \mathbf{1} \mathbf{A}_{\parallel} & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \mathbf{1} \mathbf{A}_{\parallel} \end{bmatrix}$$

and therefore

$$\text{diag}(\mathbf{A}_{\parallel} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{A}_{\parallel}) = [0, \lambda_1 - \lambda_2, \lambda_1 - \lambda_3, \dots, \lambda_r - \lambda_{r-1}, 0]^T.$$

Building upon these notions, one can seek \mathbf{S} as the solution

to the following convex problem:

$$\begin{aligned}
& \underset{\mathbf{F}, \mathbf{S}_{\parallel}, \mathbf{S}_{\perp}}{\text{minimize}} && \|\mathbf{F} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{F}\|_{\star} + \|\mathbf{S}_{\perp} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{S}_{\perp}\|_{\star} \\
& \text{s.t:} && (\mathbf{S})_{n,n'} = 0 \text{ if } (v_n, v_{n'}) \notin \mathcal{E}, n, n' = 1, \dots, N \\
& && \mathbf{S} = \mathbf{S}_{\parallel} + \mathbf{S}_{\perp} \\
& && \mathbf{S}_{\perp} = \mathbf{S}_{\perp}^T \\
& && \mathbf{S}_{\parallel} = \mathbf{S}_{\parallel}^T \\
& && \mathbf{S}_{\parallel} = \mathbf{U}_{\parallel} \mathbf{F} \mathbf{U}_{\parallel}^T \\
& && \text{tr}(\mathbf{F}) = r, \quad \text{tr}(\mathbf{S}_{\perp}) \leq N - r - \epsilon \\
& && \mathbf{S}_{\perp}^T \mathbf{U}_{\parallel} = \mathbf{0}
\end{aligned} \tag{10}$$

where $\epsilon > 0$ is a small constant and where $\|\mathbf{F} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{F}\|_{\star}$ replaces $\|\mathbf{S}_{\parallel} \otimes \mathbf{I}_N - \mathbf{I}_N \otimes \mathbf{S}_{\parallel}\|_{\star}$ since its value is the same but the size of \mathbf{F} is generally smaller than the size of \mathbf{S}_{\parallel} . To avoid matrix solutions with all zero elements, the trace of \mathbf{F} is constrained to be equal to the dimension of the subspace. Moreover, the constraint $\text{tr}(\mathbf{S}_{\perp}) \neq N - r - \epsilon$ is introduced to force the non-zero eigenvalues of \mathbf{S}_{\parallel} to differ from those of \mathbf{S}_{\perp} . If it turns out that introducing this constraint, the resulting feasible set is empty, one can replace it with $\text{tr}(\mathbf{S}_{\perp}) \geq N - r + \epsilon$. If both these constraints lead to empty feasible sets, then the projection \mathbf{P} cannot be exactly implemented by means of a graph filter for the given \mathcal{G} . It is remarkable to note that the set of topologies that allow implementation of a given \mathbf{P} through a graph filter is generally larger than the set of topologies that allow the implementation through the method of [3].

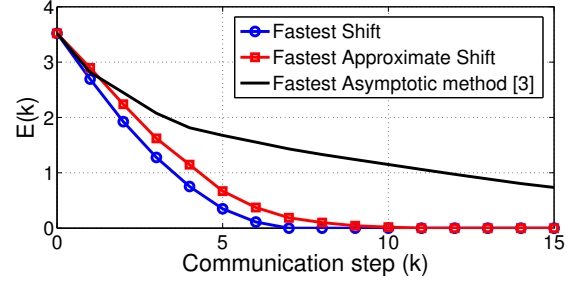
In addition to the solution given in (10), which focuses on fastest convergence, an additional approach to approximate the fastest shift with a lower level of complexity is proposed next. This approach obtains shift matrices that yield filters of a potentially higher order than the previous approach but the optimization problem has less computational complexity. The following approximate objective function is used in (10):

$$\underset{\mathbf{F}, \mathbf{S}_{\parallel}, \mathbf{S}_{\perp}}{\text{minimize}} \quad \|\mathbf{F} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{F}\|_{\star} + \|\mathbf{S}_{\perp}\|_2$$

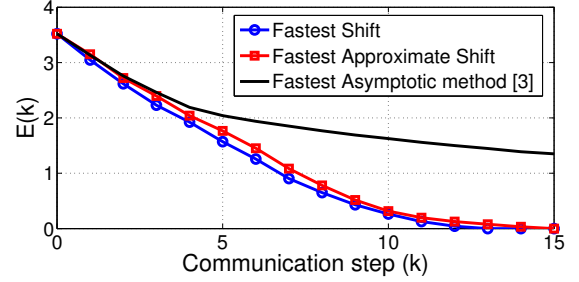
where we have replaced $\|\mathbf{S}_{\perp} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{S}_{\perp}\|_{\star}$ with $\|\mathbf{S}_{\perp}\|_2$ to minimize the largest eigenvalue of \mathbf{S}_{\perp} and thereby, to reduce the spread of eigenvalues associated with eigenvectors perpendicular to the subspace spanned by the columns of \mathbf{U}_{\parallel} . As a consequence of the term $\|\mathbf{S}_{\perp}\|_2$, the largest eigenvalue of the optimal \mathbf{S}_{\perp} typically has a higher multiplicity, yet smaller eigenvalues will typically have unit multiplicity.

IV. SIMULATION RESULTS AND DISCUSSION

This section illustrates the performance of the proposed approach by averaging the results over 100 different random networks composed of $N = 25$ nodes. At each realization of the experiment, a network was generated by deploying the N nodes uniformly at random over a square area of side R . An edge between two nodes was created if the distance between them was shorter than a predefined maximum transmission range r_{\max} . Random input signals \mathbf{z} were drawn from a zero mean and unit variance normal distribution. At every realization \mathbf{U}_{\parallel} was randomly generated by applying Gram Schmidt to a $N \times r$ matrix whose entries are independent and uniformly distributed between 0 and 1.



(a) $N = 25, r = 5, r_{\max} = 0.5R$



(b) $N = 25, r = 10, r_{\max} = 0.75R$

Fig. 1: Performance of the proposed graph filter approach vs existing fastest asymptotic approach [3].

The performance of the filter was evaluated by comparing the filtered signal $\mathbf{H}\mathbf{z}$ with the desired projected signal $\mathbf{P}\mathbf{z}$. Fig. 1 compares the error $\|\mathbf{y} - \mathbf{P}\mathbf{z}\|_2$ of the proposed approach with the one resulted from applying [3] for $N = 25$ and $r = 5, r = 10$ respectively. The value for r_{\max} was chosen to accommodate the set of feasible topologies in [3]. More specifically, the error is defined as $E(k) = \mathbb{E}_{\mathbf{A}, \mathbf{z}} \|\sum_{l=0}^k c_l^{(k)} \mathbf{S}^l \mathbf{z} - \mathbf{P}\mathbf{z}\|_2$ for the proposed algorithms and as $E(k) = \mathbb{E}_{\mathbf{A}, \mathbf{z}} \|\mathbf{W}^k \mathbf{z} - \mathbf{P}\mathbf{z}\|_2$ for the approach in [3]. For graph filter approaches, the filter coefficients $c_l^{(k)}$ that provide the fastest convergence were calculated as explained in Sec. III when $k \geq L$, and as the solution to the least squares problem $\text{vec}(\mathbf{P}) = [\text{vec}(\mathbf{I}), \text{vec}(\mathbf{S}), \dots, \text{vec}(\mathbf{S}^k)] \mathbf{c}^{(k)}$ for $k < L$, where $\mathbf{c}^{(k)} = [c_0^{(k)}, \dots, c_k^{(k)}]^T$. According to Fig. 1a and Fig. 1b, the fastest shift converges to the desired projection result in the smallest number of local exchanges. The fastest approximation approach follows closely while the fastest asymptotic approach [3] does not converge within the considered iteration limit. These observations reaffirm the expected merits of the proposed graph filter based approach, which minimizes the number of communication steps.

V. CONCLUSIONS

This paper presents a decentralized method to compute subspace projections in a minimal number of iterations. The approach relies on the notion of graph filtering and solves convex programs involving nuclear norm minimization over a judiciously designed feasible set. Simulation tests showcase the benefits of the proposed schemes. Future research will deal with improved optimization algorithms to solve (10) as well as a more exhaustive simulation study.

REFERENCES

- [1] X. Insausti, P. M. Crespo, and B. Beferull-Lozano, "In-network computation of the transition matrix for distributed subspace projection," in *Proc. International Conference on Distributed Computing in Sensor Systems*, May 2012, pp. 124–131.
- [2] J. C. Chin, I. H. Hou, J. C. Hou, C. Ma, N. S. Rao, M. Saxena, M. Shankar, Y. Yang, and D. K. Y. Yau, "A sensor-cyber network testbed for plume detection, identification, and tracking," in *Proc. International Symposium on Information Processing in Sensor Networks*, April 2007, pp. 541–542.
- [3] S. Barbarossa, G. Scutari, and T. Battisti, "Distributed signal subspace projection algorithms with maximum convergence rate for sensor networks with topological constraints," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2009, pp. 2893–2896.
- [4] F. Camaro-Nogues, D. Alonso-Roman, C. Asensio-Marco, and B. Beferull-Lozano, "Reducing the observation error in a wsn through a consensus-based subspace projection," in *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*. IEEE, 2013, pp. 3643–3648.
- [5] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, May 2013.
- [6] A. Sandryhaila, S. Kar, and J. M. F. Moura, "Finite-time distributed consensus through graph filters," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2014, pp. 1080–1084.
- [7] S. Segarra, A. G. Marques, and A. Ribeiro, "Optimal Graph-Filter Design and Applications to Distributed Linear Network Operators," *IEEE Trans. Signal Processing*, vol. 65, no. 15, pp. 4117–4131, Aug. 2017.
- [8] S. Safavi and U. A. Khan, "Revisiting finite-time distributed algorithms via successive nulling of eigenvalues," *IEEE Signal Processing Letters*, vol. 22, no. 1, pp. 54–57, Jan 2015.
- [9] S. Segarra, A. G. Marques, G. Mateos, and A. Ribeiro, "Network topology inference from spectral templates," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 3, pp. 467–483, 2017.
- [10] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," vol. 25, no. 2, pp. 21–30, 2008.