FAST DISTRIBUTED SUBSPACE PROJECTION VIA GRAPH FILTERS

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

Intelligent Signal Processing and Wireless Networks Laboratory (WISENET)
Department of ICT, University of Agder, Grimstad, Norway
Email: {thilina.weerasinghe, daniel.romero, cesar.asensio, baltasar.beferull}@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, thereby reducing congestion and increasing the robustness and the scalability of the 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.

Index Terms— Graph filter, subspace projection, WSN

1. INTRODUCTION

Wireless sensor networks (WSNs) frequently perform inference tasks in applications demanding distributed monitoring [1] and operation [2]. Many of these tasks, such as least squares estimation, denoising, weighted consensus, and distributed detection [3], [4] to name a few, can be cast as projecting the observed signal onto a subspace known to contain the true signal. Although subspace projections can be computed in a centralized manner if all nodes send their measurements to a fusion center, such a scheme is neither robust against node failures nor scalable in general. These limitations motivate distributed algorithms for subspace projection.

In the distributed subspace projection method in [5], every node obtains each iterate as a linear combination of the previous iterate of its neighbors. The weights of this linear combination are adjusted to achieve fast asymptotic convergence. An extension is proposed in [3], where the aforementioned weights are obtained in a distributed manner. Unfortunately, these approaches only give rise to asymptotic convergence 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 reduced set of topologies, as shown in [6]. These limitations have been alleviated for average consensus (a special case of subspace projection) in [7] first and later in the literature of graph signal processing [8] through graph filters [9], [10], [11]. Graph filters generalize classical time-invariant filters to accommodate signals defined on the vertices of a graph. Remarkably, these approaches for distributed average consensus converge in a finite number of iterations at the expense of introducing memory. More general scenarios of subspace projection have been addressed in [10] and [12], 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 their appplicability.

This paper develops a method to obtain not only the graph filter coefficients but also the shift matrices that enable distributed subspace projection through graph filters in a finite number of iterations. The proposed method approximately minimizes the order of the resulting filters, and therefore the number of communications between connected nodes.

The remainder of the paper is structured as follows. Sec. 2 reviews some background related to subspace projection and graph filters before formulating the problem. Sec. 3 describes the proposed algorithm. Finally, Sec. 4 presents numerical results and Sec. 5 concludes the paper.

Notation: The spectral radius of a matrix $\mathbf{A}$ is $\rho(\mathbf{A}) \triangleq \max\{|\lambda_1|, ..., |\lambda_n|\}$, being $\lambda_1, ..., \lambda_n$ its eigenvalues. The 2-norm is $||\mathbf{A}||_2 = \sqrt{\lambda_{\max}(\mathbf{A}^T\mathbf{A})} = \sigma_{\max}(\mathbf{A})$, where $\lambda_{\max}$ (respectively $\sigma_{\max}$) is the largest eigenvalue (singular value). $||\mathbf{A}||_\star$ denotes the nuclear norm of $\mathbf{A}$, $\mathsf{cols}(\mathbf{A})$ the set of its columns, $\mathcal{R}(\mathbf{A})$ the span of its columns, and $\mathsf{evals}(\mathbf{A})$ the set of its eigenvalues. Finally, $\otimes$ denotes the Kronecker product.

2. PROBLEM FORMULATION

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

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.

2.1. Distributed Subspace Projection

Consider a graph $\mathscr{G}(\mathscr{V},\mathscr{E})$, where the vertices in $\mathscr{V} = \{v_1,\dots,v_N\}$ represent N sensors and there is an edge $(v_n,v_{n'})$ in $\mathscr{E} \subset \mathscr{V}\times\mathscr{V}$ iff sensors v_n and $v_{n'}$ can communicate directly. This includes all self loops, i.e., $(v_n,v_n)\in\mathscr{E}$, $n=1,\dots,N$. Communications are assumed symmetric, which implies that $\mathscr{G}$ is undirected, i.e. $(v_n,v_{n'})\in\mathscr{E}$ iff $(v_{n'},v_n)\in\mathscr{E}$. Let $\mathbf{A}$ denote the adjacency matrix of $\mathscr{G}$, where $(\mathbf{A})_{n,n'}=1$ if $(v_n,v_{n'})\in\mathscr{E}$, and $(\mathbf{A})_{n,n'}=0$ otherwise. The goal is to estimate a certain signal vector $\boldsymbol{x}\in\mathbb{R}^N$, which quantifies the phenomenon of interest (e.g. temperature), from the observation vector $\mathbf{z}=[z_1,\dots,z_N]^T=\boldsymbol{x}+\boldsymbol{\zeta}$, where $z_n\in\mathbb{R}$ denotes the observation of node $v_n\in\mathscr{V}$ and $\boldsymbol{\zeta}\in\mathbb{R}^N$ stands for additive noise.

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

$$\hat{\boldsymbol{x}}\triangleq[\hat{x}_1,\dots,\hat{x}_N]^T=\mathbf{U}_{\|}\mathbf{U}_{\|}^T\mathbf{z}\triangleq\mathbf{P}\mathbf{z}, \tag{1}$$

where $\mathbf{P}\in\mathbb{R}^{N\times N}$ is the projection matrix. The subspace projection problem is to find $\boldsymbol{x}$ given $\mathbf{z}$ and $\mathbf{U}_{\|}$. Since the dimension r of the signal subspace is smaller than the dimension N of the observation space, $\hat{\boldsymbol{x}}$ is expected to be a better estimate of $\boldsymbol{x}$ than $\mathbf{z}$, an effect known as noise reduction.

A distributed scheme for subspace projection is proposed in [5], where the iterates $\mathbf{z}[k+1]=\mathbf{W}\mathbf{z}[k]$ are computed for $k=0,1,\dots$ with initialization $\mathbf{z}[0]=\mathbf{z}$. This scheme seeks a matrix $\mathbf{W}\in\mathbb{R}^{N\times N}$ satisfying (i) $(\mathbf{W})_{n,n'}=0$ if $(v_n,v_{n'})\notin\mathscr{E}$ and (ii) $\lim_{k\to\infty}\mathbf{z}[k]=\lim_{k\to\infty}\mathbf{W}^k\mathbf{z}=\mathbf{P}\mathbf{z}$, $\forall\mathbf{z}\in\mathbb{R}^N$. Note that the latter condition imposes that $\lim_{k\to\infty}\mathbf{W}^k=\mathbf{P}$. The number of local exchanges k required to attain a low error $||\mathbf{z}[k]-\mathbf{P}\mathbf{z}||$ is generally high since this approach only provides asymptotic convergence. Moreover, the set of feasible topologies for which conditions (i) and (ii) can be simultaneously satisfied is limited [6]. In contrast, the present paper develops an approach that yields convergence to the exact projection in a finite and nearly minimal number of iterations and for a larger set of feasible topologies.

2.2. 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 $(v_n,v_{n'})\notin\mathscr{E}$, and is referred to as the shift matrix [10]. Examples of shift matrices include $\mathbf{A}$ and $\mathbf{L}\triangleq\mathrm{diag}[\mathbf{A}\mathbf{1}]-\mathbf{A}$. It is important to notice that this shift operator can be evaluated distributedly, since the n-th entry of $\mathbf{y}\triangleq\mathbf{S}\mathbf{z}$ can be expressed as

$$y_n=\sum_{n'=1}^{N}(\mathbf{S})_{n,n'}z_{n'}=\sum_{n':(v_n,v_{n'})\in\mathscr{E}}(\mathbf{S})_{n,n'}z_{n'}, \tag{2}$$

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. 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 an operator $\mathbf{H}\mathbf{z}\mapsto\mathbf{z}$, where $\mathbf{H}\in\mathbb{R}^{N\times N}$ is a polynomial of degree $L-1$ of the graph shift operator $\mathbf{S}$

$$\mathbf{H}:=\sum_{l=0}^{L-1}c_l\mathbf{S}^l, \tag{3}$$

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

This paper develops an efficient methodology to obtain subspace projections in a distributed fashion using graph filters. To this end, $\{c_l\}_{l=0}^{L-1}$ and $\mathbf{S}$ satisfying $\mathbf{H}=\mathbf{P}$ must be found. In contrast to [5], which provides $\lim_{k\to\infty}\mathbf{W}^k=\mathbf{P}$, a graph filter provides $\mathbf{H}=\mathbf{P}$ in a finite number of iterations. This is at the expense of introducing memory in the computations, as required to linearly combine $\{\mathbf{z}[k]\}_{k=0}^{l}$.

In [9], graph filters were applied to finite-time distributed average consensus, which is a special case of subspace projection that arises by setting $\mathbf{P}=\mathbf{1}\mathbf{1}^T/N$. Moreover, in [10], a general framework to implement arbitrary linear operators using graph filters is developed and applied to consensus and network coding. However, both approaches rely on the knowledge of a suitable shift matrix, but a methodology to obtain this matrix for general subspace projection remains an open problem and is the focus of this paper. The problem can thus be formulated as: given $\mathbf{U}_{\|}$ and $\mathscr{E}$, find $\mathbf{S}\in\mathbb{R}^{N\times N}$ and $\{c_l\}_{l=0}^{L-1}$ such that $\mathbf{P}=\sum_{l=0}^{L-1}c_l\mathbf{S}^l$ with L as small as possible and $(\mathbf{S})_{n,n'}=0$ if $(v_n,v_{n'})\notin\mathscr{E}$, $n,n'=1,\dots,N$.

3. SHIFT MATRICES FOR FAST DISTRIBUTED SUBSPACE PROJECTION

This section develops an approach to obtain the shift matrix and the corresponding filter coefficients that provide exact subspace projection in an approximately minimal number of iterations. For this purpose, we first characterize the set of feasible shift matrices for a given $\mathcal{E}$ and $\mathbf{U}_{\|}$. Subsequently, the corresponding filter coefficients are computed. Finally, an optimization methodology is proposed to minimize the order of the filter, i.e. the number of communication steps needed to obtain the projection via graph filtering.

[1] To simplify notation, assume that $\mathbf{S}^0=\boldsymbol{I}$ even if $\mathbf{S}$ is not invertible.

3.1. Feasible Shifts and Filter Coefficients

In order to compute the filter coefficients, let us first characterize the set of feasible shift matrices, which is given by

$$\mathcal{S} = \Big\{ \mathbf{S} \in \mathbb{R}^{N\times N} : \mathbf{S} = \mathbf{S}^T,\ (\mathbf{S})_{n,n'} = 0 \text{ if } (v_n, v_{n'}) \notin \mathscr{E},$$
$$\exists L, \mathbf{c} = [c_0, ..., c_{L-1}]^T \text{ satisfying } \mathbf{U}_\parallel \mathbf{U}_\parallel^{\mathrm{T}} = \sum_{l=0}^{L-1} c_l \mathbf{S}^l \Big\}. \quad (4)$$

Proposition 1. *Let $\mathbf{U}_\parallel \in \mathbb{R}^{N\times r}$ have orthonormal columns and let $\mathcal{S}$ be given by* (4)*. Then,*

$$\mathcal{S} = \Big\{ \mathbf{S} \in \mathbb{R}^{N\times N} :\ (\mathbf{S})_{n,n'} = 0 \textit{ if } (v_n, v_{n'}) \notin \mathscr{E},$$
$$\mathbf{S} = \mathbf{U}_\parallel \mathbf{E} \mathbf{\Lambda}_\parallel \mathbf{E}^T \mathbf{U}_\parallel^T + \mathbf{V}_\perp \mathbf{\Lambda}_\perp \mathbf{V}_\perp^T \textit{ for some } \mathbf{E} \in \mathbb{R}^{r\times r},$$
$$\mathbf{\Lambda}_\parallel \in \mathbb{R}^{r\times r} \textit{ diagonal and invertible}, \mathbf{\Lambda}_\perp \in \mathbb{R}^{N-r\times N-r} \textit{ diagonal},$$
$$\mathbf{V}_\perp \in \mathbb{R}^{N\times N-r} \textit{satisfying } \mathbf{E}\mathbf{E}^T = \mathbf{I},\ \mathbf{V}_\perp^T \mathbf{V}_\perp = \mathbf{I},\ \mathbf{U}_\parallel^T \mathbf{V}_\perp = \mathbf{0},$$
$$(\mathbf{\Lambda}_\parallel)_{n,n} \neq\ (\mathbf{\Lambda}_\perp)_{n'n'}, \forall n \in \{1,\dots,r\}, \forall n' \in \{1,\dots,N-r\} \Big\}$$
$$= \Big\{ \mathbf{S} \in \mathbb{R}^{N\times N} :\ (\mathbf{S})_{n,n'} = 0 \textit{ if } (v_n, v_{n'}) \notin \mathscr{E},$$
$$\mathbf{S} = \mathbf{S}_\parallel + \mathbf{S}_\perp \textit{ for some } \mathbf{S}_\parallel \textit{ and } \mathbf{S}_\perp \textit{ satisfying}$$
$$\mathbf{S}_\parallel = \mathbf{S}_\parallel^T, \mathcal{R}(\mathbf{U}_\parallel) = \mathcal{R}(\mathbf{S}_\parallel), \mathbf{S}_\perp = \mathbf{S}_\perp^T, \mathsf{cols}(\mathbf{S}_\perp) \perp \mathbf{U}_\parallel,$$
$$\textit{if } \lambda \in \mathsf{evals}(\mathbf{S}_\parallel) \textit{ and } \lambda \neq 0 \textit{ then } \lambda \notin \mathsf{evals}(\mathbf{S}_\perp) \Big\}. \quad (5)$$

The proof is omitted here due to space restrictions. This proposition essentially establishes that a shift matrix $\mathbf{S}$ satisfies (3) 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 a subspace of 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

$$\mathbf{U}_\parallel \mathbf{U}_\parallel^{\mathrm{T}} = \mathbf{U}_\parallel \mathbf{E} \Big[\sum_{l=0}^{L-1} c_l \mathbf{\Lambda}_\parallel^l \Big] \mathbf{E}^T \mathbf{U}_\parallel^{\mathrm{T}} + \mathbf{V}_\perp \Big[\sum_{l=0}^{L-1} c_l \mathbf{\Lambda}_\perp^l \Big] \mathbf{V}_\perp^{\mathrm{T}}.$$

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

$$\begin{bmatrix} \mathbf{1}_r \\ \mathbf{0}_{N-r} \end{bmatrix} = \begin{bmatrix} 1 & \lambda_1 & \dots & \lambda_1^{L-1} \\ 1 & \lambda_2 & \dots & \lambda_2^{L-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_N & \dots & \lambda_N^{L-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{L-1} \end{bmatrix}, \quad (6)$$

where $\lambda_1, \dots, \lambda_N$ are such that $\mathbf{\Lambda}_\parallel \triangleq \mathrm{diag}\{\lambda_1, \dots, \lambda_r\}$ and $\mathbf{\Lambda}_\perp \triangleq \mathrm{diag}\{\lambda_{r+1}, \dots, \lambda_N\}$. Equation (6) can be expressed as:

$$\boldsymbol{\lambda}_p = \mathbf{\Psi} \mathbf{c} \quad (7)$$

which provides a means to obtain the coefficients $\{c_l\}_{l=0}^{L-1}$.

3.2. Minimization of the Filter Order

This section finds the matrix $\mathbf{S} \in \mathcal{S}$ that satisfies (3) with a nearly minimal order L. To this end, note that $\mathbf{\Psi}$ is Vandermonde and therefore its rank is $\min(L, N')$, where N' is the number of distinct eigenvalues in $\{\lambda_n\}_{n=0}^{N-1}$. Since $N' \leq N$, (7) is always satisfied for some $\mathbf{c}$ if $\mathbf{S} \in \mathcal{S}$ and $L = N$. Then, one can set $L = N$ w.l.o.g. and minimize the filter order by finding $\mathbf{c}$ and $\mathbf{S}$ such that the last entries of $\mathbf{c}$ are zero.

If $\lambda_n \neq \lambda_{n'}$ for all $n \neq n'$, then $\mathbf{\Psi}$ is invertible and $\mathbf{c} = \mathbf{\Psi}^{-1} \boldsymbol{\lambda}_p$, where 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. In contrast, if there are repeated 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 $\tilde{\mathbf{\Psi}}$ by replacing the first row of $\mathbf{\Psi}$ with $[0, 0, \dots, 0, 1]$, and form the vector $\tilde{\boldsymbol{\lambda}}_p$ by replacing the first entry of $\boldsymbol{\lambda}_p$ with a zero. The resulting $\mathbf{c} = \tilde{\mathbf{\Psi}}^{-1} \tilde{\boldsymbol{\lambda}}_p$ satisfies (7) and $c_{N-1} = 0$, which leads to a filter of order $N-1$. Similarly, if there are multiple replicated eigenvalues in $\{\lambda_n\}_{n=1}^{N}$, one can repeat the previous procedure to ensure that several entries of $\mathbf{c}$ are zero. It can be seen that the order of the fastest filter for a given $\mathbf{S}$ 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. The rest of the section accomplishes this task.

The goal is to minimize the number of different eigenvalues of $\mathbf{S} = \mathbf{S}_\parallel + \mathbf{S}_\perp$ subject to $\mathbf{S} \in \mathcal{S}$ (cf. (5)). Unfortunately, this problem is non-convex since (i) its objective is non-convex and (ii) due to the condition in (5) that requires the non-zero eigenvalues of $\mathbf{S}_\parallel$ and $\mathbf{S}_\perp$ to be different. In order to alleviate the non-convexity of the objective, one can replace it with a convex surrogate. To this end, note that if the condition mentioned in (ii) holds, the number of distinct non-zero eigenvalues of $\mathbf{S}$ is equal to the sum of the number of distinct non-zero eigenvalues of both $\mathbf{S}_\parallel$ and $\mathbf{S}_\perp$. Focusing on $\mathbf{S}_\parallel$, note that the larger the number of distinct non-zero eigenvalues of $\mathbf{S}_\parallel$, the larger 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 [13], which is proportional to $||\mathbf{\Lambda}_\parallel \otimes \mathbf{I}_N - \mathbf{I}_N \otimes \mathbf{\Lambda}_\parallel||_\star$, since by definition of nuclear norm[2], it equals the l_1-norm of the singular values of its argument. To see this, note that

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

which implies that its diagonal entries are then given by

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

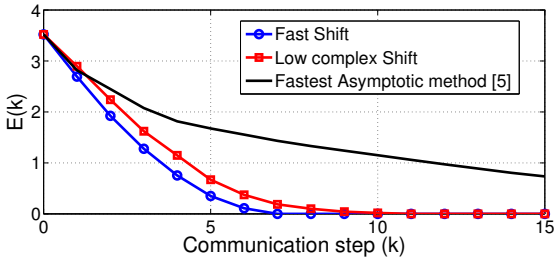
[2]The nuclear norm of a matrix $\mathbf{A}$ is $||\mathbf{A}||_\star = \mathrm{tr}(\sqrt{\mathbf{A}^*\mathbf{A}}) = \sum_i \sigma_i(\mathbf{A})$.

Building upon these notions, one can seek $\mathbf{S}$ as the solution to the following convex problem:

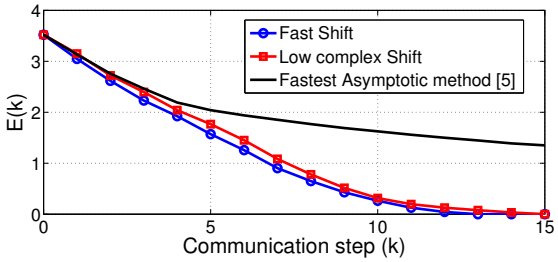
$$\begin{aligned} \underset{\mathbf{F},\mathbf{S},\mathbf{S}_{\parallel},\mathbf{S}_{\perp}}{\text{minimize}} \quad & ||\mathbf{F}\otimes\mathbf{I}-\mathbf{I}\otimes\mathbf{F}||_{\star}+||\mathbf{S}_{\perp}\otimes\mathbf{I}-\mathbf{I}\otimes\mathbf{S}_{\perp}||_{\star} \\ \text{s. t.} \quad & (\mathbf{S})_{n,n'}=0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

where $\epsilon > 0$ is a small positive constant, $\mathbf{F}$ substitutes $\mathbf{E}\mathbf{\Lambda}_{\parallel}\mathbf{E}^{T}$, and $||\mathbf{F}\otimes\mathbf{I}-\mathbf{I}\otimes\mathbf{F}||_{\star}$ replaces $||\mathbf{\Lambda}_{\parallel}\otimes\mathbf{I}_N-\mathbf{I}_N\otimes\mathbf{\Lambda}_{\parallel}||_{\star}$ since they are equal and the former avoids eigenvalue computation. The constraint $\mathsf{tr}(\mathbf{F})=r$ is introduced to avoid the trivial solution where $\mathbf{S}=\mathbf{0}$. A constraint $\mathsf{tr}(\mathbf{S}_{\perp})\neq N-r$ would in principle be needed to avoid the trivial solution in which all non-zero eigenvalues of $\mathbf{S}_{\parallel}$ are equal to one. Since such a constraint would result in a non-convex problem, one can solve instead first with the constraint $\mathsf{tr}(\mathbf{S}_{\perp})\leq N-r-\epsilon$, then with $\mathsf{tr}(\mathbf{S}_{\perp})\geq N-r+\epsilon$, and finally select the solution of these two problems yielding the smallest objective. If both these constraints lead to empty feasible sets for all $\epsilon > 0$, then the projection $\mathbf{P}$ cannot be exactly computed by means of a graph filter for the given topology. It is remarkable to note that the set of topologies that allow implementation of a particular $\mathbf{P}$ through a graph filter is generally larger than the set of topologies that allow the implementation through the method of [5]. Too see this, note that in [5] a total of $N-r$ eigenvalues of $\mathbf{W}$ have to be equal to 1 and the rest smaller than 1, which is not required here. Finally, the constraint that the non-zero eigenvalues of $\mathbf{F}$ differ from those of $\mathbf{S}_{\perp}$ is omitted since it would render the problem non-convex. In the unlikely case that a non-zero eigenvalue of $\mathbf{F}$ equals a non-zero eigenvalue of $\mathbf{S}_{\perp}$, one can solve the problem again for a different ϵ, or weighting the second term in the objective by a positive number different from 1.

In the sequel, we propose an additional approach to obtain shift matrices that yield filters of potentially higher order, but with significantly less computational complexity. To this end, the objective function of our original problem can be replaced with $||\mathbf{F}\otimes\mathbf{I}-\mathbf{I}\otimes\mathbf{F}||_{\star}+||\mathbf{S}_{\perp}||_2$, where we have replaced $||\mathbf{S}_{\perp}\otimes\mathbf{I}-\mathbf{I}\otimes\mathbf{S}_{\perp}||_{\star}$ with $||\mathbf{S}_{\perp}||_2$. The reason is that while the former requires the evaluation of all eigenvalues of its argument, the latter only requires to evaluate the largest one. The term $||\mathbf{S}_{\perp}||_2$ promotes solutions having a small spread of the eigenvalues associated to eigenvectors perpendicular to the subspace spanned by the columns of $\mathbf{U}_{\parallel}$. For this reason, the largest eigenvalue of the optimal $\mathbf{S}_{\perp}$ typically has a high multiplicity.



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



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

Fig. 1: Performance of our approach vs the one in [5].

4. SIMULATION RESULTS AND DISCUSSION

This section illustrates the performance of the proposed approach by averaging the results over 100 different random networks of N nodes. Topologies were generated by deploying the nodes uniformly at random over a square area of side R and connecting them if the internode distance was smaller than $r_{\max}$. Random input signals $\mathbf{z}$ were drawn from a normal distribution with zero mean and unit variance. At every realization $\mathbf{U}_{\parallel}$ was randomly generated by applying Gram Schmidt to an $N\times r$ matrix whose entries are independent and uniformly distributed between 0 and 1.

The performance of the filter was evaluated by comparing the filtered signal $\mathbf{Hz}$ with the desired projected signal $\mathbf{Pz}$. Fig. 1 compares the error $||\mathbf{y}-\mathbf{Pz}||_2$ of both the exact and the approximate solutions with the error resulting from applying [5] for $r=5$ and $r=10$. More specifically, the error is defined as $E(k)=\mathbb{E}_{\mathbf{A},\mathbf{z}}||\sum_{l=0}^{k}c_l^{(k)}\mathbf{S}^l\mathbf{z}-\mathbf{Pz}||_2$ for the two objectives proposed and as $E(k)=\mathbb{E}_{\mathbf{A},\mathbf{z}}||\mathbf{W}^k\mathbf{z}-\mathbf{Pz}||_2$ for the approach in [5]. According to Fig. 1a and Fig. 1b, the proposed shifts converge to the desired projection in a nearly minimal number of steps, outperforming the work in [5].

5. CONCLUSIONS

This paper presents a decentralized method to compute subspace projections in a nearly 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 scheme.

6. REFERENCES

[1] C. Chen, J. Yan, N. Lu, Y. Wang, X. Yang and X. Guan, "Ubiquitous monitoring for industrial cyber-physical systems over relay-assisted wireless sensor networks," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 3, pp. 352–362, 2015.

[2] C. Asensio-Marco and B. Beferull-Lozano, "Energy efficient consensus over complex networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 2, pp. 292–303, 2015.

[3] 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*, pp., 124–131.

[4] J. C. Chin, I. H. Hou, J. C. Hou, C. Ma and N. S. Rao, M. Saxena and M. Shankar and 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*, pp., 541–542.

[5] 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*, pp., 2893–2896.

[6] 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 *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, pp., 3643–3648.

[7] Y. A. Kibangou, "Graph laplacian based matrix design for finite-time distributed average consensus," in *American Control Conference (ACC)*. IEEE, pp., 1901–1906.

[8] D. I. Shuma, S. K. Narang and 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.

[9] 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*, pp., 1080–1084.

[10] 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.

[11] 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.

[12] 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.

[13] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.