Convergence Analysis of Distributed PAST Based on Consensus Propagation

Carolina Reyes, Robert Dallinger and Markus Rupp
Vienna University of Technology, Institute of Telecommunications, Austria
{creyes, rdalling, mrupp}@nt.tuwien.ac.at

***Abstract*—In previous work, we have analyzed the convergence properties of the so-called Projection Approximation Subspace Tracking algorithm (PAST) by means of Singular Value Decomposition methods. Based on this, we extend our analysis and study the global convergence properties of two distributed variants of the PAST algorithm based on average consensus. We provide an insight regarding convergence in the mean, and establish step-size bounds that guarantee the stability of the algorithm. Finally, we investigate the algorithms behavior at several step-sizes by means of simulation experiments.**

***Index Terms*—PAST algorithm, distributed subspace tracking, blind source separation, average consensus, step-size bounds.**

I. INTRODUCTION

Subspace tracking algorithms have become quite popular as relatively simple methods to detect subspaces, separate them and even track them. Nevertheless, most of these approaches are based on singular value or eigenvalue decomposition of large data matrices, which automatically poses the drawback of a high computational load.

Bin Yang introduced the Projection Approximation Subspace Tracking (PAST) algorithm [1], [2] using a new signal subspace model interpretation that is based on an unconstrained minimization function. The subspace tracking is implemented by employing the recursive least squares algorithm, which relies on an appropriate projection approximation. This algorithm solves the subspace estimation problem in a centralized setting, where all the information is available at one single processing entity. In contrast, this work focusses on situations where the information is not available at a central processing unit, e.g., in a wireless sensor network.

In [3], we introduced a distributed version of the PAST algorithm and presented its performance analysis in [4]. Here, we deal with two distributed subspace tracking methods and present a convergence analysis based on [5].

The PAST algorithm minimizes the cost function in a Least-Squares (LS) sense, i.e.,

$$J = \min_{\mathbf{W}} E\left[\left\|\underline{\mathbf{x}} - \mathbf{W}\mathbf{W}^H\underline{\mathbf{x}}\right\|^2\right]. \tag{1}$$

Here, $\underline{\mathbf{x}} \in \mathbb{C}^{N\times 1}$ is the data observed at time k and $\mathbf{W} \in \mathbb{C}^{N\times r}$ is the signal subspace containing r narrow-band signal waves that impinge an array of N sensors. Starting with initial values $\widehat{\mathbf{R}}_{yy,0} = \mathbf{I}$ and $\mathbf{W}_0 \in \mathbb{C}^{N\times r} = [\mathbf{I}_r, \mathbf{0}]^T$, a recursive algorithm is applied to a sequence of vectors $\underline{\mathbf{x}}_k$ in order to iteratively solve the minimization problem in (1) using the continuously incoming observations $\underline{\mathbf{x}}_k$:

$$\underline{\mathbf{y}}_k = \mathbf{W}_{k-1}^H\underline{\mathbf{x}}_k \tag{2}$$

$$\underline{\mathbf{e}}_k = \underline{\mathbf{x}}_k - \mathbf{W}_{k-1}\mathbf{W}_{k-1}^H\underline{\mathbf{x}}_k \tag{3}$$

$$\widehat{\mathbf{R}}_{yy,k} = \widehat{\mathbf{R}}_{yy,k-1} + \alpha_k\left[\underline{\mathbf{y}}_k\underline{\mathbf{y}}_k^H - \widehat{\mathbf{R}}_{yy,k-1}\right] \tag{4}$$

$$\mathbf{W}_k = \mathbf{W}_{k-1} + \gamma_k\underline{\mathbf{e}}_k\underline{\mathbf{y}}_k^H\left(\widehat{\mathbf{R}}_{yy,k}\right)^{-1}. \tag{5}$$

The so-called subspace update equation in (5) applies the inverse of the estimated autocorrelation matrix $\widehat{\mathbf{R}}_{yy,k} = E\left\{\underline{\mathbf{y}}_k\underline{\mathbf{y}}_k^H\right\}$. This is typically achieved by the matrix inversion lemma to reduce the computational complexity. Other fast variants are possible to derive and have been proposed (see [6]) but are out of the scope of this paper. We restrict to a relatively general form of the algorithm with two free parameters α_k and γ_k. The primer is bounded by $0 < \alpha_k \leq 1$ and acts as forgetting factor for the decaying memory estimation of the autocorrelation matrix $\widehat{\mathbf{R}}_{yy,k}$. A value of α_k close to one gives more emphasis to the most recent rank-one update $\underline{\mathbf{y}}_k\underline{\mathbf{y}}_k^H$ and leads to a faster decaying memory. A value close to zero, in turn, results in only slight changes of $\widehat{\mathbf{R}}_{yy,k}$. While the optimal choice of α_k depends on the tracking problem, the choice of γ_k directly affects the convergence behavior of the algorithm which gives reason to consider it more thoroughly in this work. This paper stems on the lines presented in [5] for the centralized PAST algorithm. However, it extends them in order to investigate the behavior of two decentralized subspace tracking methods which are both based on the singular value decomposition.

This paper is organized as follows: Section II is devoted to the description of two algorithmic approaches and to the statement of our assumptions. A first order analysis investigating the algorithmic behavior in the mean is provided in Section III. Section IV corroborates our findings through simulation results. Finally, some concluding remarks round the paper up in Section V.

II. DESCRIPTION OF THE ALGORITHMS

To decentralize the PAST algorithm, we regard (2)-(5) as the primary set of equations running locally at the same time in each node i. This requires us to extend all the variables occurring in (2)-(5) by a localization index i, i.e., $\underline{\mathbf{y}}_{i,k} \in \mathbb{C}^{r\times 1}, \widehat{\mathbf{R}}_{i,k}^{yy} \in \mathbb{C}^{r\times r}$, $\underline{\mathbf{e}}_{i,k} \in \mathbb{C}^{|\mathcal{N}_i|\times 1}, \underline{\mathbf{x}}_{i,k} \in \mathbb{C}^{|\mathcal{N}_i|\times 1}$, $\mathbf{W}_{i,k} \in \mathbb{C}^{|\mathcal{N}_i|\times r}$ as well as $\alpha_{i,k}$ and $\gamma_{i,k}$. Furthermore, we introduce the neighborhood $\mathcal{N}_i$, which is defined as the ordered set containing the indices of all nodes that are direct

neighbors of i, including node i itself.

We moreover assume that the communication among the nodes is done synchronously, each node i broadcasts its own observation $\{x_{i,k}\}$ and receives $\{x_{j,k}\}_{\substack{\forall j\in\mathcal{N}_i \\ j\neq i}}$. With this available information, every node i aggregates a local observation vector $\underline{\mathbf{x}}_{i,k}$, which from a global point of view can be expressed in terms of the global observation vector $\underline{\mathbf{x}}_k$, according to

$$\underline{\mathbf{x}}_{i,k} = \mathbf{S}_i^T \underline{\mathbf{x}}_k. \tag{6}$$

Here, we introduced the selection matrix $\mathbf{S}_i$, a truncated adjacency matrix with dimension $(N \times |\mathcal{N}_i|)$ that consists of the columns of all $j \in \mathcal{N}_i$. This is defined as

$$(S_i)_{l,j} = \begin{cases} 1 & \text{if } j = l\text{th node} \in \mathcal{N}_i \\ 0 & \text{otherwise} \end{cases}.$$

From (2) we know that $\underline{\mathbf{y}}_{i,k} = \mathbf{W}_{i,k-1}^H \underline{\mathbf{x}}_{i,k}$ and substituting (6) for $\underline{\mathbf{x}}_{i,k}$ yields $\underline{\mathbf{y}}_{i,k} = \mathbf{W}_{i,k-1}^H \mathbf{S}_i^T \underline{\mathbf{x}}_k$. Furthermore, we introduce the matrix $\mathbf{V}_{i,k-1}^H = \mathbf{W}_{i,k-1}^H \mathbf{S}_i^T$ and let the original subspace spanned by the columns of $\mathbf{W}_{i,k}$ be embedded in the matrix $\mathbf{V}_{i,k} \in \mathbb{C}^{N\times r}$. Under such considerations, the local signal update vector turns into

$$\underline{\mathbf{y}}_{i,k} = \mathbf{V}_{i,k-1}^H \underline{\mathbf{x}}_k. \tag{7}$$

We consider the average consensus algorithm [7], a method which allows to compute the average of a given deterministic variable in a distributed fashion, and propose the two following distributed schemes:

1. Node i sends $\{\underline{\mathbf{y}}_{i,k}\}$ to its neighborhood $\mathcal{N}_i$ and receives $\{\underline{\mathbf{y}}_{j,k}\}_{\substack{\forall j\in\mathcal{N}_i \\ j\neq i}}$. For the first iteration, we let the initial values $\underline{\mathbf{y}}_{i,0} = \underline{\mathbf{y}}_{i,-1}$. For the time iterations $k > 1$, the averaging is performed according to

$$\underline{\mathbf{y}}_{i,k}(t) = g_{ii}\underline{\mathbf{y}}_{i,k}(t-1) + \sum_{\substack{j\in\mathcal{N}_i \\ j\neq i}} g_{ij}\underline{\mathbf{y}}_{j,k}(t-1). \tag{8}$$

Here, g_{ii} stands for the weight of node i and g_{ij} is the weight associated to neighbor j from $\mathcal{N}_i$. The elements $g_{i,j}$ form the weight matrix $\mathbf{G}$, as proposed in [8]. Furthermore, $t = 1, \ldots, t_{\max}$[1] is the averaging time window used for each node in order to reach a global consensus, e.g.

$$\widetilde{\underline{\mathbf{y}}}_{i,k} = \underline{\mathbf{y}}_{i,k}(t_{\max}). \tag{9}$$

2. In the second approach, node i sends the correlation matrix $\{\widehat{\mathbf{R}}_{i,k}^{yy}\}$ to all neighboring nodes j and receives $\{\widehat{\mathbf{R}}_{j,k}^{yy}\}_{\substack{\forall j\in\mathcal{N}_i \\ j\neq i}}$. The first iteration $\widehat{\mathbf{R}}_{i,0}^{yy} = \widetilde{\mathbf{R}}_{i,-1}^{yy}$, and for $k > 1$, the averaging occurs as long as

$$\widehat{\mathbf{R}}_{i,k}^{yy}(t) = g_{ii}\widehat{\mathbf{R}}_{i,k}^{yy}(t-1) + \sum_{\substack{j\in\mathcal{N}_i \\ j\neq i}} g_{ij}\widehat{\mathbf{R}}_{j,k}^{yy}(t-1). \tag{10}$$

[1] $t_{\max}$: time when consensus is reached.

The global consensus is achieved when

$$\widetilde{\mathbf{R}}_{i,k}^{yy} = \widehat{\mathbf{R}}_{i,k}^{yy}(t_{\max}). \tag{11}$$

The goal of this paper is to evaluate the convergence properties of these distributed algorithms. To achieve this, we need to find a global description that integrates (8), respectively (10), together with the localized versions of (2)-(5) running at each node. A block diagonal representation for the global algorithms allocates the local variables at the diagonal entries of the block matrices as

$$\begin{array}{ll} \underline{\mathbf{y}}_{i,k} \longrightarrow \mathbf{Y}_k \in \mathbb{C}^{rN\times N} & \widetilde{\underline{\mathbf{y}}}_{i,k} \longrightarrow \widetilde{\mathbf{Y}}_k \in \mathbb{C}^{rN\times 1} \\ \widehat{\mathbf{R}}_{i,k}^{yy} \longrightarrow \widehat{\mathbf{R}}_k^{yy} \in \mathbb{C}^{rN\times rN} & \widetilde{\mathbf{R}}_{i,k}^{yy} \longrightarrow \widetilde{\mathbf{R}}_k^{yy} \in \mathbb{C}^{rN\times r} \\ \underline{\mathbf{e}}_{i,k} \longrightarrow \mathbf{E}_k \in \mathbb{C}^{\sum_{i=1}^N |\mathcal{N}_i|\times N} & \mathbf{V}_{i,k} \longrightarrow \mathbf{V}_k \in \mathbb{C}^{N^2\times rN} \\ \mathbf{S}_i \longrightarrow \mathbf{S} \in \{0,1\}^{N^2\times\sum_{i=1}^N|\mathcal{N}_i|}. & \end{array}$$

The global descriptions of the distributed-PAST algorithms are shown in Algorithm 1 and Algorithm 2.

Input: $\alpha_k, \gamma_k, \widehat{\mathbf{R}}_0^{yy}, \mathbf{V}_0$
for $k := 1, 2, \ldots$ do
  Input: $\underline{\mathbf{x}}_k$
  $\mathbf{Y}_k = \mathbf{V}_{k-1}^H (\underline{\mathbf{I}}_N \otimes \underline{\mathbf{x}}_k)$
  $\widetilde{\mathbf{Y}}_k = (\mathbf{G}^{t_{\max}} \otimes \mathbf{I}_r)\, \mathbf{V}_{k-1}^H (\underline{\mathbf{x}}_k \otimes \underline{\mathbf{1}}_N)$
  $\mathbf{E}_k = \mathbf{S}^T [(\mathbf{I}_N \otimes \underline{\mathbf{x}}_k) - \mathbf{V}_{k-1} (\mathbf{G}^{t_{max}} \otimes \mathbf{I}_r)\, \mathbf{Y}_k]$
  $\widehat{\mathbf{R}}_k^{yy} = \widehat{\mathbf{R}}_{k-1}^{yy} + \alpha \left(\widetilde{\mathbf{Y}}_k \widetilde{\mathbf{Y}}_k^H - \widehat{\mathbf{R}}_{k-1}^{yy} \right)$
  $\mathbf{V}_k = \mathbf{V}_{k-1} + \gamma_k \mathbf{S}\mathbf{E}_k\, diag\left(\widetilde{\mathbf{Y}}_k^H\right)\left(\widehat{\mathbf{R}}_k^{yy}\right)^{-1}$ (12)
end

Algorithm 1. Distributed PAST using Equation (8).

Input: $\alpha_k, \gamma_k, \widehat{\mathbf{R}}_0^{yy}, \mathbf{V}_0$
for $k := 1, 2, \ldots$ do
  Input: $\underline{\mathbf{x}}_k$
  $\mathbf{Y}_k = \mathbf{V}_{k-1}^H (\mathbf{I}_N \otimes \underline{\mathbf{x}}_k)$
  $\mathbf{E}_k = \mathbf{S}^T \left(I_{N^2} - \mathbf{V}_{k-1}\mathbf{V}_{k-1}^H\right) (\underline{\mathbf{I}}_N \otimes \underline{\mathbf{x}}_k)$
  $\widehat{\mathbf{R}}_k^{yy} = \widehat{\mathbf{R}}_{k-1}^{yy} + \alpha \left(\mathbf{Y}_k \mathbf{Y}_k^H - \widehat{\mathbf{R}}_{k-1}^{yy} \right)$
  $\widetilde{\mathbf{R}}_k^{yy} = (\mathbf{G}^{t_{\max}} \otimes \mathbf{I}_r)\, \widehat{\mathbf{R}}_k^{yy}\, (\mathbf{1}_N \otimes \mathbf{I}_r)$
  $\mathbf{V}_k = \mathbf{V}_{k-1} + \gamma_k \mathbf{S}\mathbf{E}_k \mathbf{Y}_k^H\ diag\left(\widetilde{\mathbf{R}}_k^{yy}\right)^{-1}$ (13)
end

Algorithm 2. Distributed PAST using Equation (10).

We acknowledge that in Algorithm 1, the update equation $\widetilde{\mathbf{Y}}_k = \left[\underline{\mathbf{y}}_{1,k}, \ldots, \underline{\mathbf{y}}_{N,k}\right]$ is indeed containing the averaged signal values from each node i. Note that for updating (12), we restructure $\widetilde{\mathbf{Y}}_k$ in sucha a way that it becomes a block diagonal structure. The same situation occurs in Algorithm 2, where $\widetilde{\mathbf{R}}_k^{yy} = \left[\widetilde{\mathbf{R}}_{1,k}^{yy}, \ldots, \widetilde{\mathbf{R}}_{N,k}^{yy}\right]^T$. Here, an update of (13) requires a reshaping of the global correlation matrix $\widetilde{\mathbf{R}}_k^{yy}$. A more detailed derivation of both algorithms is omitted due to lack of space, but we can already observe that the update equation from (12) and (13) has the same structure as the signal subspace update (5) from the centralized solution in [5].

As a result, it is straightforward to apply the guidelines from [5] to evaluate the convergence properties of both distributed algorithms.

III. ANALYSIS IN THE MEAN

In this chapter we provide a first order analysis for Algorithm 1 and Algorithm 2. The main goal is to obtain a unitary matrix for the $\mathbf{W}_{i,k}$ embedded in $\mathbf{V}_k$, i.e., $\lim_{k\to\infty} \mathbf{W}_{i,k}^H \mathbf{W}_{i,k} = \mathbf{I}_r$. We assume that the data vector $\underline{\mathbf{x}}_k$ containing all the observations in the network, is of random nature and has an autocorrelation matrix $\mathbf{R}_k^{xx} = E\left\{ (\underline{\mathbf{x}}_k \otimes \underline{\mathbf{1}}_N)(\underline{\mathbf{x}}_k \otimes \underline{\mathbf{1}}_N)^H \right\}$. We are interested in a first order analysis of the signal subspace update of Algorithm 1 and Algorithm 2. Therefore, we evaluate the mean of (12) and rewrite it as

$$\begin{aligned} E\{\mathbf{V}_k\} &= E\{\mathbf{V}_{k-1}\} \\ &+ \gamma_k E\left\{ (\mathbf{I}_{N^2} - \mathbf{V}_{k-1}\mathbf{V}_{k-1}^H)\underline{\mathbf{x}}_k \underline{\mathbf{x}}_k^H \mathbf{V}_{k-1} \right\} \\ &\times E\left\{ \left(\widehat{\mathbf{R}}_k^{yy} \right)^{-1} \right\}. \end{aligned} \tag{14}$$

Similarly, (13) is expressed as

$$\begin{aligned} E\{\mathbf{V}_k\} &= E\{\mathbf{V}_{k-1}\} \\ &+ \gamma_k E\left\{ (\mathbf{I}_{N^2} - \mathbf{V}_{k-1}\mathbf{V}_{k-1}^H)\underline{\mathbf{x}}_k \underline{\mathbf{x}}_k^H \mathbf{V}_{k-1} \right\} \\ &\times E\left\{ \left(\widetilde{\mathbf{R}}_k^{yy} \right)^{-1} \right\}. \end{aligned} \tag{15}$$

In order to solve the two above equations, it is nevertheless necessary to consider the iterative approximations $\left(\widetilde{\mathbf{R}}_k^{yy}\right)^{-1}$ and $\left(\widehat{\mathbf{R}}_k^{yy}\right)^{-1}$ to be independent of the signal subspace matrix $\mathbf{V}_k$. We model $\underline{\mathbf{x}}_k$ as a random, ergodic process and let the calculation of the ensemble average $\mathbf{R}_k^{yy}$ of the given process be determined from its time average $\widehat{\mathbf{R}}_k^{yy}$ over past samples as

$$\mathbf{R}_k^{yy} = E\left\{ \widehat{\mathbf{R}}_k^{yy} \right\} = E\left\{ \mathbf{Y}_k \mathbf{Y}_k^H \right\} = \mathbf{V}_{k-1}^H \mathbf{R}_k^{xx} \mathbf{V}_{k-1},$$

and its matrix decomposition yields

$$\mathbf{R}_k^{yy} = \mathbf{P}_1 \bar{\mathbf{\Sigma}}_k \mathbf{\Lambda}_k^{xx} \bar{\mathbf{\Sigma}}_k \mathbf{P}_1^H.$$

Due to this averaging process, it is a plausible assumption to regard such average values as independent of each other. That is to say, we allow them to be statistically independent [9], and let the decomposition of the autocorrelation matrix $\mathbf{R}^{xx} = \mathbf{Q}\mathbf{\Lambda}^{xx}\mathbf{Q}^H$. To continue, we recall from [5] that a steady-state solution only exists if $E\{\mathbf{V}_k\} = [\mathbf{U}_1, \mathbf{U}_2][\mathbf{\Sigma}_k, \mathbf{0}]^T [\mathbf{P}_1 \mathbf{P}_2]^H$ and $\mathbf{U}_1 = \mathbf{Q}$. We applied an SVD to (14) and (15) and partitioned it into the significant part $\mathbf{U}_1 \mathbf{\Sigma}_k \mathbf{P}_1^H$ and a zero block. While diagonalizing these equations, we observe that all block matrices encompass the same left block matrix $\mathbf{U}_1$ and the right block matrix $\mathbf{P}_1^H$. These matrices are simplified in such way that we finally arrive at the center block matrix $\mathbf{\Sigma}_k$ comprising $\mathbf{\Sigma}_{i,k}$ diagonal elements, i.e.,

$$\begin{aligned} \mathbf{\Sigma}_k &= \mathbf{\Sigma}_{k-1} + \gamma_k (\mathbf{I} - \mathbf{\Sigma}_{k-1}\mathbf{\Sigma}_{k-1}) \mathbf{\Lambda}^{xx} \mathbf{\Sigma}_{k-1} \\ &\times (\bar{\mathbf{\Sigma}}_k \mathbf{\Lambda}^{xx} \bar{\mathbf{\Sigma}}_k)^{-1}. \end{aligned} \tag{16}$$

The above expression is obtained with (14). If (15) is used instead, an analogous expression can be found.

Furthermore, it is possible to revise (16) in terms of the local singular values at each node i. That is to say

$$\sigma_{i,k}^l = \sigma_{i,k-1}^l + \gamma_{i,k} \frac{\left(1 - \left(\sigma_{i,k-1}^l\right)^2\right) \sigma_{i,k-1}^l}{\left(\bar{\sigma}_{i,k}^l\right)^2},$$

where $l = 1, \ldots, r$. Here, $\bar{\sigma}_{i,k-1}^l$ are the singular values calculated by diagonalizing the time average $\left(\widehat{\mathbf{R}}_k^{yy}\right)^{-1}$. They should not be confused with the instantaneous singular values $\sigma_{i,k}^l$ derived from decomposing $\mathbf{V}_k$. Under the assumption that all signals are perfectly decorrelated, the steady state solution is achieved if, i.e., $\sigma_{i,k-1}^l \to 1$. Thus:

$$1 - \sigma_{i,k}^l = (1 - \sigma_{i,k-1}^l) \left[1 - \gamma_k \frac{(1 + \sigma_{i,k-1}^l)\sigma_{i,k-1}^l}{\left(\bar{\sigma}_{i,k}^l\right)^2} \right].$$

As all singular values are positive we can again see that the PAST algorithm converges in the mean for a sufficiently small step-size $\gamma_{i,k} > 0$ if $\frac{(1+\sigma_{i,k-1}^l)\sigma_{i,k-1}^l}{\left(\bar{\sigma}_{i,k}^l\right)^2}$ is bounded. In [5], the following more conservative upper bound for $\sigma_{i,k-1}^l$ was found

$$0 < \gamma_{i,k} < \frac{2\left(\bar{\sigma}_{i,k}^l\right)^2}{0.25 + 2\|\mathbf{W}_{i,k-1}^H \mathbf{W}_{i,k-1}\|^2} = \gamma_{\max_{i,k}} \left(\bar{\sigma}_{\min_{i,k}}^l\right)^2. \tag{17}$$

The knowledge of the smallest singular value $\bar{\sigma}_{\min_{i,k}}^l$ is decisive for convergence. Nevertheless, calculating it or even finding an estimate for it would lead to a high computational burden. Therefore, we follow the guidelines from [5] and alter the $\mathbf{V}_k$ update equation in Algorithm 1 and Algorithm 2 such that calculating the smallest singular value becomes nonessential. Accordingly, (12) becomes

$$\mathbf{V}_k = \mathbf{V}_{k-1} + \gamma_k \mathbf{S}\mathbf{E}_k \mathbf{A}, \tag{18}$$

and (13) yields

$$\mathbf{V}_k = \mathbf{V}_{k-1} + \gamma_k \mathbf{S}\mathbf{E}_k \mathbf{B}. \tag{19}$$

Recall the different step-sizes originally introduced in [5], which guarantee feasible step-sizes bounds and let them be addressed at each node i as

- PAST-I:

$$\mathbf{A} = diag\left(\widetilde{\mathbf{Y}}_k^H\right) \left(\widehat{\mathbf{R}}_k^{yy}\right)^{-1} \left(\bar{\sigma}_{\min_{i,k}}^l\right)^2$$
$$\mathbf{B} = \mathbf{Y}_k^H diag\left(\widetilde{\mathbf{R}}_k^{yy}\right)^{-1} \left(\bar{\sigma}_{\min_{i,k}}^l\right)^2$$

- PAST-II:

$$\mathbf{A} = diag\left(\widetilde{\mathbf{Y}}_k^H\right) \left(\widehat{\mathbf{R}}_k^{yy}\right)^{-1} \mathbf{R}_{\mathbf{V}_{i,k}}$$
$$\mathbf{B} = \mathbf{Y}_k^H diag\left(\widetilde{\mathbf{R}}_k^{yy}\right)^{-1} \mathbf{R}_{\mathbf{V}_{i,k}}$$

$$0 < \gamma_{I_{i,k}} = \gamma_{II_{i,k}} < \gamma_{\max_{i,k}} \leq \min_l \frac{2}{\left(1 + \sigma_{i,k-1}^l\right)\sigma_{i,k-1}^l} \tag{20}$$

- PAST-III:

$$\mathbf{A} = diag\left(\widetilde{\mathbf{Y}}_k^H\right)\left[\widehat{\mathbf{R}}_k^{yy} + \beta\mathbf{I}_r\right]^{-1}$$
$$\mathbf{B} = \mathbf{Y}_k^H\left[diag\left(\widetilde{\mathbf{R}}_k^{yy}\right) + \beta\mathbf{I}_r\right]^{-1}$$

$$0 < \gamma_{III_{i,k}} < \gamma_{\max_{i,k}}\delta_{\min_{i,k}} \leq \min_l \frac{2\left(\left(\bar{\sigma}_{i,k-1}^l\right)^2 + \delta\right)}{\left(1+\sigma_{i,k-1}^l\right)\sigma_{i,k-1}^l} \quad (21)$$

- PAST-IV:

$$\mathbf{A} = diag\left(\widetilde{\mathbf{Y}}_k^H\right)\mathbf{I}_r \qquad \mathbf{B} = \mathbf{Y}_k^H\mathbf{I}_r$$

$$0 < \gamma_{IV_{i,k}} < \frac{\gamma_{\max_{i,k}}}{\mathrm{tr}\left(\mathbf{R}_{i,k}^{xx}\right)} \leq \min_l \frac{2}{\left(1+\sigma_{i,k-1}^l\right)\sigma_{i,k-1}^l\lambda_i^{l,xx}}. \quad (22)$$

For the PAST-I version, we assume knowledge of the smallest singular value only. This can be achieved by tailored tracking algorithms, e.g., [10]. The PAST-II variant calculates an average $\mathbf{R}_{\mathbf{V},k}$ for the matrix $\mathbf{V}_{k-1}^H\mathbf{V}_{k-1}$, given by

$$\mathbf{R}_{\mathbf{V},k} = \mathbf{R}_{\mathbf{V},k-1} + \alpha_k(\mathbf{V}_{k-1}^H\mathbf{V}_{k-1} - \mathbf{R}_{\mathbf{V},k-1}), \quad (23)$$

and is applied to compensate for the inverse singular values of $\widehat{\mathbf{R}}_k^{yy}$ or $\widetilde{\mathbf{R}}_k^{yy}$. The PAST-III scheme introduces a regularization for computing the inverse of this matrix, which prevents numerical issues due to possible ill conditioned matrices. Finally, the PAST-IV version can be considered as the gradient-type version of the original PAST algorithm from [2]. In order to avoid the eigenvalue decomposition of the local data correlation matrix $\mathbf{R}_{i,k}^{xx} = E\{\underline{\mathbf{x}}_{i,k}\underline{\mathbf{x}}_{1,k}\}$ for finding $\lambda_i^{l,xx}$ in (22), we consider only its trace. We also let $0 < \beta \leq 1$, $\delta = \beta/\lambda_i^{l,xx}$ and $\delta_{\min} = \beta/\lambda_{\max_i}^{xx}$. Note that the choice of β also influences the convergence speed. The higher the selected value, the faster the achieved convergence. To summarize, by utilizing the aforementioned step-sizes we evade calculating the singular values in $\widehat{\mathbf{R}}_k^{yy}$ or $\widetilde{\mathbf{R}}_k^{yy}$, as would be required in (17). Furthermore, the computation of a safe upper bound of $\gamma_{i,k}$ relies only on the matrix norm, which is a computationally feasible operation. As shown in [5], the induced matrix 2-norm provides tight results. Finally, with these feasible step-sizes, an evaluation of the algorithmic behavior is presented in the next section.

IV. SIMULATION RESULTS

The following experiments apply both distributed PAST variants to a typical beam steering scenario. Consider a wireless sensor network with $N = 12$ nodes, where each sensor i observes the data $\underline{\mathbf{x}}_{i,k}$ at each time k given by

$$\underline{\mathbf{x}}_{i,k} = \sum_{l=1}^{r} \underline{\mathbf{a}}(\theta_l)s_{l,k} + \underline{\mathbf{v}}_k = \mathbf{A}\underline{\mathbf{s}}_k + \mathbf{v}_k,$$

We use $r = 3$ different angles of arrivals set to $\theta = \{0.01, 0.03, 0.2\}$ radians. Furthermore, $\mathbf{A} \in \mathbb{C}^{|\mathcal{N}_i|\times r}$ is a deterministic signal mixing matrix, $\underline{\mathbf{s}}_k$ represents the source vector and $\underline{\mathbf{v}}_k$ is the additive noise uncorrelated with $\underline{\mathbf{s}}_k$. It is possible to model a single element of this matrix as

$$a_{l,i} = \exp\left[-j\frac{2\pi}{\lambda}(\xi_i\cos\theta_l + \eta_i\sin\theta_l)\right],$$

$\forall l = 1,\ldots,r$ and $\forall i = 1,\ldots,N$. The variables (ξ_i, η_i) represent the Cartesian coordinates of the i-th sensor node in the array. The bounds (20)-(22) define a time-variant maximal value $\gamma_{i,k}^{\max}$ and we select fractions $\alpha\gamma_{i,k}^{\max}$, where $\alpha \in (0, 1]$. We further set $\alpha_k = \alpha$ in (4) by which we reduce one degree of freedom and let $\gamma_k = \alpha\gamma_{\max,k}$. We choose the maximum number of consensus iterations to be $t_{\max} = 100$ and we average over 100 Monte Carlo (MC) runs. Note that for the centralized versions we expect the algorithms to work for values α from zero to one.

We implement Algorithm 1 in a regular topology, which means that every node i in the network has the same degree $|\mathcal{N}_i| = 4$. Figure 1 depicts the distance measure $\sum_{l=1}^{r}\left(1-\sigma_k^l\right)$ being averaged over all nodes in the network. Here, the original update equation from (12), known as Distributed PAST, shows the fastest convergence followed by the PAST-I and PAST-II variants, which need smaller step-sizes to achieve the same steady-state. The PAST-III and PAST-IV schemes perform considerably slower and during the simulations for step-size values of α above 0.9, they both show first signs of instability.

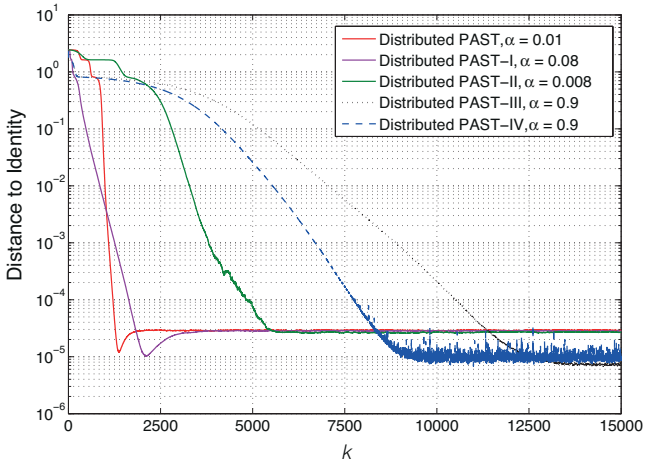

Fig. 1. Averaged distance to identity for Algorithm 1 with a regular network topology.

Figure 2 provides more results for the same simulation settings as those used for Figure 1. However, for this experiment we implemented a network topology which is random irregular. The former means that the $\mathcal{N}_i$ selected neighborhood is not necessarily composed of the j sensors closest to i, they are actually selected in a random fashion. By irregular we refer to different node degrees in the network, e.g., $|\mathcal{N}_i| = 3, 5$, and 8. Note that for all MC runs we use the irregular topology that was generated in the first simulation. We move on to Figure 2 and immediately recognize that PAST-III shows a faster convergence, in contrast to all the

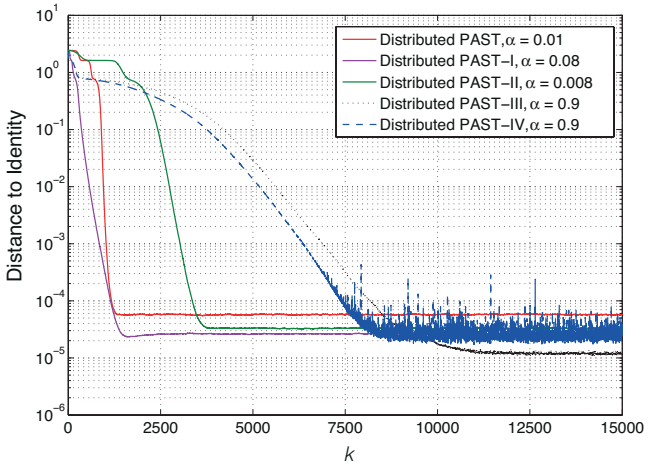

Fig. 2. Averaged distance to identity for Algorithm 1 with a random irregular network topology.

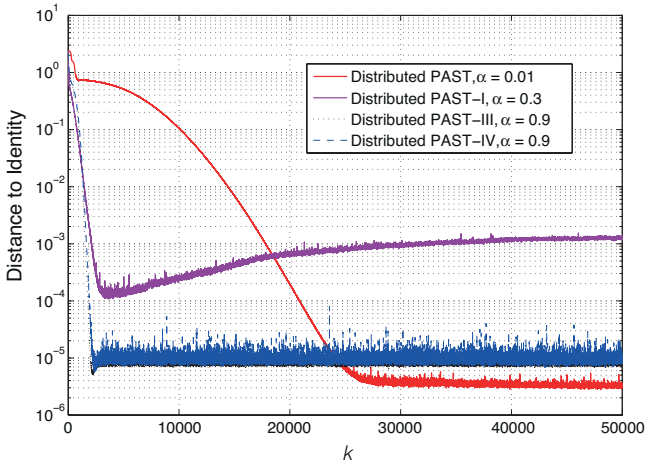

Fig. 3. Averaged distance to identity for Algorithm 2 with a regular network topology.

other variants that show practically the same performance.

Now we evaluate Algorithm 2 where the correlation matrix $\widehat{\mathbf{R}}_{i,k}^{yy}$ is exchanged between the nodes. Here, PAST-III and PAST-IV lead to the fastest, practically overlapping results for both regular (Figure 3) and irregular (Figure 4) algorithmic topologies. Again, the stability limit is found near to step-size $\alpha = 0.9$. We likewise observe that for both topologies PAST-I converges very slowly and that stability issues arise for step-sizes larger than 0.3. We do not present any plot for PAST-II, since the simulation results did not lead to a steady-state. Finally, by several experiments with both algorithms we realized how important it is to initialize the local autocorrelation matrix $\widehat{\mathbf{R}}_{i,0}^{yy}$ and the local signal subspace $\mathbf{W}_{i,k}$ by appropriate values. For both, Algorithm 1 and Algorithm 2, the initial $\widehat{\mathbf{R}}_{i,k}^{yy}$ value was defined as an identity matrix. The signal subspace $\mathbf{W}_{i,k}$ in Algorithm 1 was characterized as a truncated identity matrix of size $|\mathcal{N}_i| \times r$, for Algorithm 2 we defined it to be all ones.

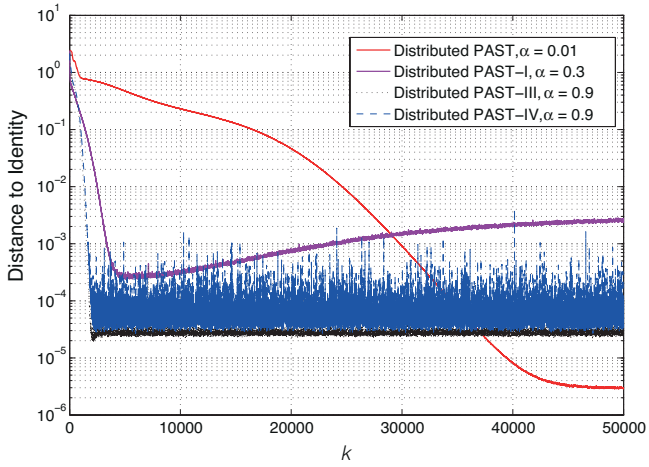

Fig. 4. Averaged distance to identity for Algorithm 2 with a random irregular network topology.

V. CONCLUSIONS

We presented two distributed subspace tracking algorithmic variants based on average consensus together with a mathematical convergence analysis that works for both approaches in the mean sense. We have likewise analyzed their behavior when the signal subspace update equation is altered in order to guarantee convergence for a range of step-sizes. Finally, we compared the results when Algorithm 1 and Algorithm 2 run in a regular and a random irregular network topology and provided satisfactory results for both cases.

ACKNOWLEDGEMENT

This work was supported by the FWF under Awards S10611-N13 within the National Research Network SISE.

REFERENCES

[1] B. Yang, "Subspace tracking based on the projection approach and the recursive least squares method," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP'93)*, Apr. 1993, vol. 4, pp. 145 –148.

[2] B. Yang, "Projection approximation subspace tracking," *IEEE Transactions in signal processing*, vol. 43, no. 1, pp. 95–107, Jan. 1995.

[3] C. Reyes, T. Hilaire, and C. F. Mecklenbräuker, "Distributed projection approximation subspace tracking based on consensus propagation," in *The 3rd International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Aruba, Dec. 2009.

[4] C. Reyes, T. Hilaire, S. Paul, and C. F. Mecklenbräuker, "Evaluation of the root mean square error performance of the PAST-consensus algorithm," in *International ITG Workshop on Smart Antennas (WSA 2010)*, Germany, Feb. 2010.

[5] M. Rupp and C. Reyes, "Robust versions of the PAST algorithm," *Proc. of EUSIPCO*, Bucharest, Rumania. Sep. 2012.

[6] K. Abed-Meraim, A. Chkeif, and Y. Hua, "Fast orthonormal past algorithm," *IEEE Signal processing letters*, vol. 7, no. 3, Mar. 2000.

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

[8] L. Xiao, S. Boyd, and S. Lall, "Distributed average consensus with time varying metropolis weights," June 2006.

[9] S. Haykin, *Adaptive Filter Theory*, Englewood Cliffs, NJ: Prentice–Hall, Inf. and System Sciences Series, 1986.

[10] Z.Bai et al., *Templates for the Solution of Algebraic Eigenvalue Problems: a Practical Guide*, SIAM, Philadelphia, 2000.