# Distributed Trace Ratio Optimization in Fully-Connected Sensor Networks

Cem Ates Musluoglu and Alexander Bertrand
*KU Leuven, Department of Electrical Engineering (ESAT),*
*STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics, Belgium*
{cemates.musluoglu, alexander.bertrand}@esat.kuleuven.be

***Abstract*—The trace ratio optimization problem consists of maximizing a ratio between two trace operators and often appears in dimensionality reduction problems for denoising or discriminant analysis. In this paper, we propose a distributed and adaptive algorithm to solve the trace ratio optimization problem over network-wide covariance matrices, which capture the spatial correlation across sensors in a wireless sensor network. We focus on fully-connected network topologies, in which case the distributed algorithm reduces the communication bottleneck by only sharing a compressed version of the observed signals at each given node. Despite this compression, the algorithm can be shown to converge to the maximal trace ratio as if all nodes would have access to all signals in the network. We provide simulation results to demonstrate the convergence and optimality properties of the proposed algorithm.**

***Index Terms*—Dimensionality reduction, distributed optimization, trace ratio, discriminant analysis, SNR optimization, wireless sensor networks.**

## I. Introduction

The trace ratio optimization (TRO) problem consists of finding a low-dimensional subspace projection, such that the total energy across all subspace dimensions of the projected data points is maximized for one data class and minimized for another one. This requirement appears in various signal processing and machine learning problems [1]–[6]. The TRO problem takes its roots from Fisher's linear discriminant [7] and the Foley-Sammon transform (FST) [8], [9]. In [10], an efficient way to compute the FST is described, which guarantees a minimum within-class and maximum between-class scatter for each single-dimensional space spanned by the individual discriminant vectors, i.e., one by one in a greedy fashion. However, due to its greedy definition, this optimal ratio between within-class and between-class scatter does not hold for the space spanned by the whole set of these vectors. This was pointed out in [11] and a method to find a generalized optimal set was proposed. However, as mentioned in [1], the method suffered from separability issues on the projected set of vectors. The same paper defined the generalized Foley-Sammon transform, which has a quotient of trace operators as the objective to maximize, which eventually has led to the TRO problem.

Algorithms solving this TRO problem have been proposed by [4] using the Grassmann manifold, [6] by semidefinite programming, and [1]–[3] using an iterative method on an auxiliary function. The original TRO is also often replaced by a generalized eigenvalue problem [12]–[14], which can again be viewed as a greedy relaxation of the TRO problem [3], which makes it akin to the greedy formulation in the original FST, while also introducing a different constraint set on the spanning vectors. As a result, a generalized eigenvalue decomposition (GEVD) does not solve the true TRO problem, but a different (yet related) greedy problem in a different constraint set.

In this paper, we study the TRO problem in a distributed setting in the context of wireless sensor networks (WSNs) where there is spatial correlation across the sensors. In this case, the TRO problem is defined by the network-wide spatial covariance matrices, which are assumed to be unknown. Such cases appear for example in body-sensor or neuro-sensor networks [15], [16] in which miniaturized sensor devices exploit spatial correlation to decode and classify neural signals (e.g left vs. right hand imaginary movement [17]). Our goal is to solve the TRO problem in such distributed settings with a reduced communication bandwidth compared to the centralized setting. While the corresponding distributed generalized eigenvalue problem has been studied in [18], the true TRO problem has not been studied in such a distributed context. In this conference contribution, we focus on fully-connected network topologies, although the results can be extended to more general topologies as well, based on similar strategies as in [19]. We propose an adaptive distributed TRO algorithm, which only exchanges compressed signals across the nodes to reduce the communication bottleneck. Although the compression is lossy, in the sense that the original signals cannot be perfectly reconstructed, it is lossless at the same time in the sense that convergence to the optimal network-wide solution of the TRO problem can be guaranteed, i.e., each node has access to the projected samples onto the TRO subspace. We provide simulations on synthetic data which demonstrate the convergence properties of our proposed method along with an empirical analysis on the convergence rate.

## II. Review of the Trace Ratio Optimization (TRO) Problem

### A. Definition and Interpretation of the TRO Problem

The TRO problem aims to find a subspace spanned by the columns of the $M \times Q$ matrix $X$ such that the following trace ratio is maximized:

$$\begin{aligned} \underset{X}{\text{maximize}} \ & \varrho(X) \triangleq \frac{\text{tr}(X^T A X)}{\text{tr}(X^T B X)} \\ \text{subject to} \ & X \in \mathcal{S}, \end{aligned} \tag{1}$$

where "tr" denotes the trace operator, $A, B$ are symmetric positive (semi-)definite[1] $M \times M$ matrices and $\mathcal{S} = \{X \in \mathbb{R}^{M \times Q} : X^T X = I_Q\}$, with $I_Q$ the $Q \times Q$ identity matrix. $X$ is the optimization variable and contains in its columns $Q$ orthonormal vectors with $Q \ll M$. Depending on the context, the matrices $A$ and $B$ can have different meanings. For example, in linear discriminant analysis, the aim is to tightly group points of a same class while separating each class from another in the best way possible [20]. Therefore, in that context, $A$ would be the within-scatter matrix and $B$ the between-scatter matrix of the data points. This method has

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 802895). The authors also acknowledge the financial support of the KU Leuven Research Council for project C14/16/057, the FWO (Research Foundation Flanders) for project G.0A49.18N, and the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme.

[1]To ensure that the maximum exists, the matrix $B$ has to satisfy some additional rank properties, which will be explained in Section II-C.

been used to learn the weights of a Mahalanobis distance in [21].

In a signal processing context, the matrices $A$ and $B$ can be viewed as covariance matrices corresponding to two stationary $M$-channel signals, denoted by $\mathbf{y}(t)$ and $\mathbf{v}(t) \in \mathbb{R}^M$. Then, $A$ and $B$ would represent $R_{\mathbf{yy}} = E[\mathbf{y}(t)\mathbf{y}(t)^T]$ and $R_{\mathbf{vv}} = E[\mathbf{v}(t)\mathbf{v}(t)^T]$ where $E[\cdot]$ denotes the expectation operator. For example, in motor imagery brain-computer interfaces based on electroencephalography (EEG), $\mathbf{y}(t)$ could represent EEG activity during imaginary left hand movement, while $\mathbf{v}(t)$ could represent EEG activity during imaginary right hand movement [17]. Solving (1) then provides a spatial filter bank with $M$ inputs and $Q$ outputs, of which the output power can then be used to discriminate between these two signal classes. In the case of signal denoising, $\mathbf{y}$ and $\mathbf{v}$ would be observed during "signal-plus-noise" segments and "noise-only" segments respectively [18], in which case $X$ would act as a denoising filter bank.

In the following parts of this text, we consider that $\varrho$ in (1) is defined such that $A = R_{\mathbf{yy}}$ and $B = R_{\mathbf{vv}}$, i.e.,

$$\begin{aligned} \underset{X}{\text{maximize}} \ & \varrho(X) \triangleq \frac{\text{tr}(X^T R_{\mathbf{yy}} X)}{\text{tr}(X^T R_{\mathbf{vv}} X)} \\ \text{subject to} \ & X \in \mathcal{S}. \end{aligned} \tag{2}$$

Moreover, we will assume that all signals are short-term stationary and ergodic with zero mean, therefore $\frac{1}{N}\sum_{t=1}^{N} \mathbf{y}(t)\mathbf{y}(t)^T \approx E[\mathbf{y}(t)\mathbf{y}(t)^T] = R_{\mathbf{yy}}$ for a convenient number of time samples $N$, and similarly for $\mathbf{v}$. We will mostly define the variables according to $\mathbf{y}$, but every definition will have its equivalent for $\mathbf{v}$. We will omit the time index $t$ for notational convenience.

### B. Comparison to Generalized Eigenvalue Methods

If $Q = 1$, then the TRO problem becomes equivalent to the generalized Rayleigh coefficient, maximized by the generalized eigenvector (GEVC) corresponding to the largest generalized eigenvalue (GEVL) of the pair $(R_{\mathbf{yy}}, R_{\mathbf{vv}})$. Results between a generalized eigenvalue problem and the TRO problem start to diverge in general when $Q > 1$, i.e., the GEVCs corresponding to the $Q$ largest GEVLs of the pair $(R_{\mathbf{yy}}, R_{\mathbf{vv}})$ do not solve the TRO problem but instead maximize (2) under another constrain set, namely $X^T R_{\mathbf{vv}} X = I_Q$. While replacing (2) with a GEVD problem is a popular strategy in the literature, there have been various arguments to opt for solving the true TRO problem (2) instead of the corresponding GEVD problem. In particular, [22] explains that enforcing orthogonality on the filters leads to higher discriminating power because these projections do not distort the metric structure. Moreover, in [4] it is shown that the GEVD solution will not necessarily result in a larger optimal value for $\varrho$, compared to the TRO case, and [2] claims that the natural way to describe the problem at hand is with the TRO formulation.

### C. Solving the TRO problem in a centralized context

Various iterative methods have been proposed to solve the TRO problem. In this paper, we focus on the method in [2], as it will be the basis for the distributed algorithm in Section III. We assume further that the rank of $R_{\mathbf{vv}}$ is strictly larger than $M - Q$ so that the denominator is non-zero $\forall X \in \mathcal{S}$ and the maximum value $\rho^*$ which $\varrho$ can assume exists and is finite, as shown in [23], where it is also explained that the maximum is obtained for $X^* \in \mathbb{R}^{M \times Q}$, unique up to a unitary transformation.

To define the iterative algorithm for solving the TRO problem, we first point out that the problem has an equivalent form, which can be seen by defining the auxiliary function $h : \mathcal{S} \times \mathbb{R} \to \mathbb{R}$ :

$$h(X, \rho) = \text{tr}(X^T(R_{\mathbf{yy}} - \rho R_{\mathbf{vv}})X). \tag{3}$$

In [1], it is shown that an optimal $X^*$ that satisfies (2) must also satisfy the following sufficient conditions:

$$\max_{X \in \mathcal{S}} h(X, \rho) = 0 \iff \rho = \rho^*, \tag{4}$$

and

$$X^* \in \arg\max_{X \in \mathcal{S}} h(X, \rho^*). \tag{5}$$

Hence, we transform the initial problem to finding the root of the function $\max_{X \in \mathcal{S}} h(X, \rho)$. For a given scalar $\rho$, this function outputs the sum of the $Q$ largest eigenvalues (EVLs) of $(R_{\mathbf{yy}} - \rho R_{\mathbf{vv}})$ and the $X$ that maximizes (3) contains the orthonormal eigenvectors (EVCs) corresponding to these EVLs:

$$(R_{\mathbf{yy}} - \rho R_{\mathbf{vv}})X = X\Lambda, \tag{6}$$

where $\Lambda$ is a diagonal matrix containing the $Q$ largest EVLs. With this knowledge, an iterative method to solve the TRO problem is given in Algorithm 1, which was originally described in [2], where also convergence to a solution of the TRO problem is proved.

**Algorithm 1:** Trace Ratio Maximization Algorithm [2]

**input :** $R_{\mathbf{yy}}, R_{\mathbf{vv}} \in \mathbb{R}^{M \times M}$
**output:** $X^*, \rho^*$

$X^0$ initialized randomly, $\rho^0 \leftarrow \varrho(X^0)$, $i \leftarrow 0$
**repeat**
  1) $X^{i+1} \leftarrow \text{EVC}_Q(R_{\mathbf{yy}} - \rho^i R_{\mathbf{vv}})$, where $\text{EVC}_Q(A)$ extracts $Q$ orthonormal eigenvectors corresponding to the $Q$ largest eigenvalues of $A$.
  2) $\rho^{i+1} \leftarrow \varrho(X^{i+1})$
  $i \leftarrow i + 1$
**until** *Convergence*

**Remark 1.** *As the solution of the TRO problem is unique up to a unitary transformation (i.e., $X^*R$ is a solution if $X^*$ is a solution and $R$ is a unitary matrix), we define the equality up to a unitary transform of the columns as $\overset{*}{=}$.*

## III. Distributed TRO algorithm in Fully-Connected WSNs

Suppose now that we have a WSN with $K$ nodes in $\mathcal{K} = \{1, \ldots, K\}$, where each node $k$ measures two $M_k$-channel signals $\mathbf{y}_k$ and $\mathbf{v}_k$. We define the network-wide signal $\mathbf{y} \in \mathbb{R}^M$, with $M = \sum_{k \in \mathcal{K}} M_k$, to be the stacked signals of the nodes, i.e., $\mathbf{y} = \left[\mathbf{y}_1^T, \ldots, \mathbf{y}_K^T\right]^T$ and similarly $\mathbf{v} = \left[\mathbf{v}_1^T, \ldots, \mathbf{v}_K^T\right]^T$. In the distributed case, every node $k$ has access to observations of its own signals $\mathbf{y}_k$ and $\mathbf{v}_k$, but not to $\mathbf{y}_l$ and $\mathbf{v}_l$, $l \in \mathcal{K} \backslash \{k\}$. Therefore, we cannot use Algorithm 1 directly because the eigenvalue decomposition (EVD) step requires the network-wide signals to estimate the network-wide spatial covariance matrices $R_{\mathbf{yy}}$ and $R_{\mathbf{vv}}$, which cannot be estimated by any single node, unless all the raw sensor signal observations would be transmitted to a fusion center, which creates a bandwidth bottleneck. Instead, we will solve the TRO problem in an adaptive and distributed fashion by letting each node only transmit compressed observations of its local signals.

In this paper, we focus on fully-connected WSNs, which implies that signal observations broadcast by any node are

received by all other nodes in the network. This allows for a more intelligible description of the distributed algorithm, but is not a limitation per se, since the analysis can be extended to other topologies using similar strategies as in [19], which is out of the scope of this paper.

Note that Algorithm 1 consists of two alternating steps, which involves solving an eigenvalue problem (step 1) followed by evaluating the trace ratio $\rho$ in the current point (step 2). A naive approach for solving (2) in a distributed setting would consist of computing the eigenvalue decomposition in step 1 using a distributed (G)EVD algorithm, e.g., the so-called DACGEE[2] [18] or DACMEE [24] algorithm. However, these algorithms are iterative themselves, so we would need to wait for them to converge and only then will we be able to update $\rho$ in step 2, and the alternation between steps 1 and 2 are then iterated as well. These hierarchically nested iterations would make convergence extremely slow. Instead, we aim for an adaptive distributed algorithm without such hierarchically nested iterations, which runs at a single time scale.

The algorithm we propose is referred to as the Distributed Trace Ratio Optimization algorithm (DTRO). We start by partitioning $X$ as:

$$X = \left[X_1^T, \ldots, X_K^T\right]^T \in \mathbb{R}^{M\times Q}, \tag{7}$$

such that $X_k \in \mathbb{R}^{M_k\times Q}\ \forall k \in \mathcal{K}$ and the objective in (2) can be rewritten as:

$$\varrho(X) = \frac{\text{tr}\left(\sum_{k,l} X_k^T R_{\mathbf{y}_k\mathbf{y}_l} X_l\right)}{\text{tr}\left(\sum_{k,l} X_k^T R_{\mathbf{v}_k\mathbf{v}_l} X_l\right)} = \frac{E\left[||\sum_k X_k^T\mathbf{y}_k||^2\right]}{E\left[||\sum_k X_k^T\mathbf{v}_k||^2\right]}, \tag{8}$$

where $k,l \in \mathcal{K}$, $R_{\mathbf{y}_k\mathbf{y}_l} = E[\mathbf{y}_k\mathbf{y}_l^T]$, $R_{\mathbf{v}_k\mathbf{v}_l} = E[\mathbf{v}_k\mathbf{v}_l^T]$. The DTRO algorithm will update $X_k$'s across the nodes in a sequential round-robin fashion, where node $k$ is responsible for updating $X_k$.

We assume that, at iteration $i$ of the DTRO algorithm, all nodes $k$ linearly compress their sensor observations by a compression matrix $F_k^i \in \mathbb{R}^{M_k\times P}$, with $P < M_k$, before broadcasting them to the other nodes of the network. This compression is done through the following linear combination:

$$\widehat{\mathbf{y}}_k^i = F_k^{iT}\mathbf{y}_k,\ \widehat{\mathbf{v}}_k^i = F_k^{iT}\mathbf{v}_k, \tag{9}$$

such that $\widehat{\mathbf{y}}_k^i, \widehat{\mathbf{v}}_k^i \in \mathbb{R}^P$. Suppose node $q$ is the updating node at iteration $i$, where node $q$ receives compressed observations of $\widehat{\mathbf{y}}_k^i, \widehat{\mathbf{v}}_k^i$ from nodes $k \in \mathcal{K}\backslash\{q\}$. The main question that arises is whether the updating node $q$ can get enough information out of those signals received, such that it can contribute to getting a closer estimation of $X^*$. In the DTRO algorithm, we will set $F_k^i = X_k^i$ (with $P = Q$). In this case, the projection of the network-wide signal $\mathbf{y}$ onto the subspace spanned by the columns of $X^i$ can then be computed as:

$$\widehat{\mathbf{y}}^i \triangleq X^{iT}\mathbf{y} = \sum_{k\in\mathcal{K}} X_k^{iT}\mathbf{y}_k = \sum_{k\in\mathcal{K}} \widehat{\mathbf{y}}_k^i \tag{10}$$

and similarly for $\mathbf{v}$. Hence, node $q$ is able to compute the global objective $\varrho(X^i)$ from its own observations and the compressed signals it receives from the other nodes. It is noted that in this case, $X^i$ acts both as a compression matrix and the variable we want to optimize. Let us assume node $q$ is the updating node at iteration $i$. Node $q$'s own sensor signals $\mathbf{y}_q$ are stacked with the compressed sensor signals of the other nodes, which results in the vector:

$$\widetilde{\mathbf{y}}_q^i = [\mathbf{y}_q^T, \widehat{\mathbf{y}}_1^{iT}, \ldots, \widehat{\mathbf{y}}_{q-1}^{iT}, \widehat{\mathbf{y}}_{q+1}^{iT}, \ldots, \widehat{\mathbf{y}}_K^{iT}]^T \tag{11}$$

[2]DACM(G)EE: distributed adaptive covariance-matrix (generalized) eigenvector estimation.

of length $\widetilde{M}_q = M_q + Q(K-1)$. At the beginning of each iteration of the DTRO algorithm, the updating node $q$ collects a contiguous stream of $N$ time samples of $\widetilde{\mathbf{y}}_q^i$ to be able to estimate the covariance matrix $R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q} = E[\widetilde{\mathbf{y}}_q^i\widetilde{\mathbf{y}}_q^{iT}] \in \mathbb{R}^{\widetilde{M}_q\times\widetilde{M}_q}$ of the information available. We note that, in each iteration in which node $q$ updates, these covariance matrices are estimated on a new stream of observations, exploiting the stationarity property. Therefore, despite the iterative nature of the algorithm, the same block of samples is never communicated twice, making the algorithm "adaptive" rather than "iterative". This then also allows to track slow changes in the signal statistics, under the condition that these are slower than the convergence rate of the algorithm.

We define a matrix $C_q^i$ that allows us to relate (11) to the network-wide signal $\mathbf{y}$ such that:

$$\widetilde{\mathbf{y}}_q^i = C_q^{iT}\mathbf{y}. \tag{12}$$

By correspondence of the elements in both variables, it can be deduced that $C_q^i = \left[\begin{array}{c|c|c} 0 & B^i_{<q} & 0 \\ \hline I_{M_q} & 0 & 0 \\ \hline 0 & 0 & B^i_{>q} \end{array}\right] \in \mathbb{R}^{M\times\widetilde{M}_q}$, where $B^i_{<q}$ and $B^i_{>q}$ are block diagonal matrices containing $X_1^i, \ldots, X_{q-1}^i$ and $X_{q+1}^i, \ldots, X_K^i$ respectively on their (block-)diagonals.

Then, the local covariance matrix at node $q$ can be expressed as $R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q} = C_q^{iT}R_{\mathbf{yy}}C_q^i$, and using the parameterization:

$$X = C_q^i\widetilde{X}_q, \tag{13}$$

we have a new local variable $\widetilde{X}_q \in \mathbb{R}^{\widetilde{M}_q\times Q}$ at node $q$. Substituting (13) in (2) allows to define the following compressed and parameterized version of the TRO problem (2), which can be solved locally at node $q$:

$$\begin{aligned} \underset{\widetilde{X}_q}{\text{maximize}}\ & \tilde{\varrho}_q^i(\widetilde{X}_q) \triangleq \frac{\text{tr}(\widetilde{X}_q^T R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q}\widetilde{X}_q)}{\text{tr}(\widetilde{X}_q^T R^i_{\widetilde{\mathbf{v}}_q\widetilde{\mathbf{v}}_q}\widetilde{X}_q)} \\ \text{subject to}\ & \widetilde{X}_q \in \widetilde{\mathcal{S}}_q^i, \end{aligned} \tag{14}$$

where $\widetilde{\mathcal{S}}_q^i = \{\widetilde{X}_q \in \mathbb{R}^{\widetilde{M}_q\times Q} : \widetilde{X}_q^T C_q^{iT} C_q^i \widetilde{X}_q = I_Q\}$. It is noted that $\varrho(X) = \tilde{\varrho}_q^i(\widetilde{X}_q)$. Due to the parameterization (13), the compressed optimization problem (14) can be viewed as the optimization of the network-wide problem (2) over $X$, but with extra constraints which constrain the variable $X_k$ with $k \neq q$ to have the same column space as $X_k^i$ (this can be seen from the definition of $C_q^i$, where the submatrices $B^i_{<q}$ and $B^i_{>q}$ contain the $X_k^i$'s, $k \neq q$, on their diagonal blocks).

We can derive from (3), (5) the local auxiliary problem:

$$\max_{\widetilde{X}_q\in\widetilde{\mathcal{S}}_q^i} \text{tr}\left(\widetilde{X}_q^T(R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q} - \rho^i R^i_{\widetilde{\mathbf{v}}_q\widetilde{\mathbf{v}}_q})\widetilde{X}_q\right), \tag{15}$$

where $\rho^i$ can be computed using node $q$'s own observations and the ones received from other nodes, following the relationships given in (8) and (10):

$$\rho^i = \frac{E\left[||X_q^{iT}\mathbf{y}_q + \sum_{k\in\mathcal{K}\backslash\{q\}}\widehat{\mathbf{y}}_k^i||^2\right]}{E\left[||X_q^{iT}\mathbf{v}_q + \sum_{k\in\mathcal{K}\backslash\{q\}}\widehat{\mathbf{v}}_k^i||^2\right]}. \tag{16}$$

Based on Algorithm 1, we should now compute the EVCs of the matrix $(R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q} - \rho^i R^i_{\widetilde{\mathbf{v}}_q\widetilde{\mathbf{v}}_q})$. However, the constraint set $\widetilde{S}_q^i$ imposes orthogonality with respect to the matrix:

$$\begin{aligned} K_q^i &= \text{Blkdiag}(I_{M_q}, L_1^i, \ldots, L_{q-1}^i, L_{q+1}^i, \ldots, L_K^i) \\ &= C_q^{iT}C_q^i, \qquad \text{with } L_k^i = X_k^{iT}X_k^i, \end{aligned} \tag{17}$$

i.e., $\widetilde{X}_q^T K_q^i \widetilde{X}_q = I_Q$. Therefore, we replaced the EVD problem (5) with the following GEVD problem, which can be solved locally at node $q$:

$$(R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q} - \rho^i R^i_{\widetilde{\mathbf{v}}_q\widetilde{\mathbf{v}}_q})\widetilde{X}_q = K_q^i \widetilde{X}_q \widetilde{\Lambda}_q, \text{ with } \widetilde{X}_q^T K_q^i \widetilde{X}_q = I_Q, \tag{18}$$

with $\widetilde{\Lambda}_q \in \mathbb{R}^{Q\times Q}$ diagonal. This is the relationship analogous to (6) in a distributed context based on the compressed observations available to the updating node $q$.

**Remark 2.** *We further assume that both matrices in the pair $(R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q} - \rho^i R^i_{\widetilde{\mathbf{v}}_q\widetilde{\mathbf{v}}_q}, K_q^i)$ are full rank and their largest $Q+1$ GEVLs are all distinct, so that the solution to (18) is well-defined. If this assumption does not hold, some technical modifications to the algorithm are necessary to ensure convergence (details omitted).*

As mentioned earlier, $X = C_q^i \widetilde{X}_q$ implies that node $q$ has only the full freedom of updating its own local compression matrix $X_q \in \mathbb{R}^{M_q\times Q}$ from (7), while the matrices $X_k$ corresponding to the other nodes $k \neq q$ are constrained to preserve their original column space. This can be seen from the following partitioning of the variable $\widetilde{X}_q$:

$$\widetilde{X}_q = \left[X_q^T, G_1^T, \ldots, G_{q-1}^T, G_{q+1}^T, \ldots, G_K^T\right]^T, \tag{19}$$

where $X_q$ corresponds to the first $M_q$ rows of $\widetilde{X}_q$ and each $G_k \in \mathbb{R}^{Q\times Q}$, such that $X = C_q^i \widetilde{X}_q$ implies that:

$$X_k^{i+1} = \begin{cases} X_q & \text{if } k = q \\ X_k^i G_k & \text{if } k \neq q \end{cases}. \tag{20}$$

Therefore, following the partition of (19), when node $q$ computes $\widetilde{X}_q$ by solving (18), node $q$ communicates to all other nodes $k \in \mathcal{K}\backslash\{q\}$ the matrices $G_k$ so that the latter can update their local variable $X_k$ according to (20). All these steps are summarized in Algorithm 2, which describes the DTRO algorithm. It is noted that, as the GEVCs, as computed in step 4, are only defined up to the signs of their columns, we may observe oscillations between the signs of the columns across iterations. Step 5 of Algorithm 2 resolves this problem by choosing the signs based on those of the previous iteration.

If one would remove step 3 of the DTRO algorithm, i.e., fix $\rho^i$ across iterations to any arbitrary value $\rho$, we obtain an instance of the DACGEE algorithm from [18] on the matrix pair $(R_{\mathbf{yy}} - \rho R_{\mathbf{vv}}, I_M)$. This shows that the DTRO algorithm actually interleaves the iterations of Algorithm 1 with the iterations of a distributed GEVD algorithm. However, note that convergence of the DTRO algorithm is not implied by the convergence of Algorithm 1, since the former does not solve the network-wide EVC in each iteration (but only partially in one of the nodes). Similarly, the convergence of DACGEE in [18] does not imply convergence of the DTRO algorithm as $\rho$ changes in each iteration, which changes the eigenvalue problem. Nevertheless, it can be shown that the DTRO algorithm also converges, as formalized in the following theorem.

**Theorem 1.** *For any initialization $X^0 \in \mathbb{R}^{M\times Q}$, the updates of Algorithm 2 satisfy $\lim_{i\to+\infty} X^i \overset{*}{=} X^*$ where $X^*$ is a solution of (2), i.e., DTRO converges to the optimal solution. In particular, it converges to the same TRO solution as Algorithm 1 up to a sign ambiguity in the columns of $X^*$.*

The proof is omitted due to space limitations and will be provided in a future extended version of the manuscript.

**Algorithm 2:** Distributed Trace Ratio Optimization

**output:** $X^*, \rho^*$

$X^0$ initialized randomly, $\rho^0 \leftarrow \varrho(X^0)$, $i \leftarrow 0$

**repeat**

- $q \leftarrow (i \mod K) + 1$
- 1) Node $q$ receives $L_k^i = X_k^{iT} X_k^i$ and $\widehat{\mathbf{y}}_k^i(t)$, $\widehat{\mathbf{v}}_k^i(t)$ for $t = iN+1, \ldots, iN+N$ from all other nodes $k \neq q$
- 2) Node $q$ estimates $R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q}, R^i_{\widetilde{\mathbf{v}}_q\widetilde{\mathbf{v}}_q}$ based on the stacking defined in (11)
- 3) Compute $\rho^i$ from (16)
- 4) $\widetilde{X}_q \leftarrow \text{GEVC}_Q(R^i_{\widetilde{\mathbf{y}}_q\widetilde{\mathbf{y}}_q} - \rho^i R^i_{\widetilde{\mathbf{v}}_q\widetilde{\mathbf{v}}_q}, K_q^i)$, where $\text{GEVC}_Q(A,B)$ extracts the $B-$orthogonal $Q$ generalized eigenvectors corresp. to the $Q$ largest generalized eigenvalues of $(A,B)$, and $K_q^i$ is given in (17)
- 5) $\widetilde{X}_q \leftarrow \widetilde{X}_q U^{i+1}$, where $U^{i+1} \in \mathcal{D}$, the set of signature matrices, i.e. diagonal matrices containing either 1 or $-1$ in their diagonals, and $U^{i+1} = \text{argmax}_{U\in\mathcal{D}} ||X_q^{i+1} U - X_q^i||_F$
- 6) Partition $\widetilde{X}_q$ as in (19), broadcast $G_k, \forall k \neq q$
- 7) Every node updates $X_k^{i+1}$ according to (20)
- $i \leftarrow i+1$

**until** *Convergence*

## IV. Experimental Results

To demonstrate our results, we consider in this section a setting similar to the one in [18]. We fix the number of nodes to $K = 50$ and the number of sensors on each node to $M_k = 15$, $\forall k \in \mathcal{K}$, hence $M = K \cdot M_k$. Then, the network-wide signal $\mathbf{y}$ is modeled as:

$$\mathbf{y}(t) = \Gamma \cdot \mathbf{d}(t) + \mathbf{v}(t), \tag{21}$$

where the noise in $\mathbf{v}$ is modeled as a combination of spatially correlated noise and spatially white noise, i.e.,

$$\mathbf{v}(t) = T \cdot \mathbf{s}(t) + \mathbf{n}(t). \tag{22}$$

$T$ is an $M \times L$ and $\Gamma$ is an $M \times Q$ matrix and their elements are drawn independently from the uniform distribution in $[-0.5, 0.5]$. The elements of $\mathbf{d} \in \mathbb{R}^Q$, $\mathbf{s} \in \mathbb{R}^L$ are drawn independently and follow a normal distribution with zero-mean and variance of 0.5, i.e., $\mathcal{N}(0, 0.5)$. The model is therefore a mixture of $L+Q$ point sources, of which $L$ are interfering sources, represented by $\mathbf{s}$, that are continuously active and $Q$ of them are the ones of interest, represented by $\mathbf{d}$ and have an on-off behaviour. Note that during the inactivity of the desired source in $\mathbf{d}$, it holds that only $\mathbf{v}$ is observed at the sensors, which allows to collect observations of both $\mathbf{y}$ and $\mathbf{v}$. The elements of additive noise given by $\mathbf{n}$ follows an independent normal distribution $\mathcal{N}(0, 0.1)$. We set $L = 5$, $Q = 5$ and the number of samples the updating node $q$ gets per iteration to $N = 10000$ for both $\mathbf{y}$ and $\mathbf{v}$. The latter is therefore the amount of samples over which the covariance matrices are estimated at each iteration, which is an arbitrary choice. In practice, the proper value of $N$ depends on the specific application, in particular on the sensor sampling rate, and the required adaptivity-vs-accuracy trade-off. Figure 1 shows the convergence results of our experiments. The results have been obtained using 200 independent Monte Carlo runs, using the same settings as precised above. In these comparisons, we estimated the network-wide (centralized) solutions $X^*$ and $\rho^*$ using Algorithm 1 in each independent run, where the stopping criterion was a threshold of $10^{-12}$ in the difference of two

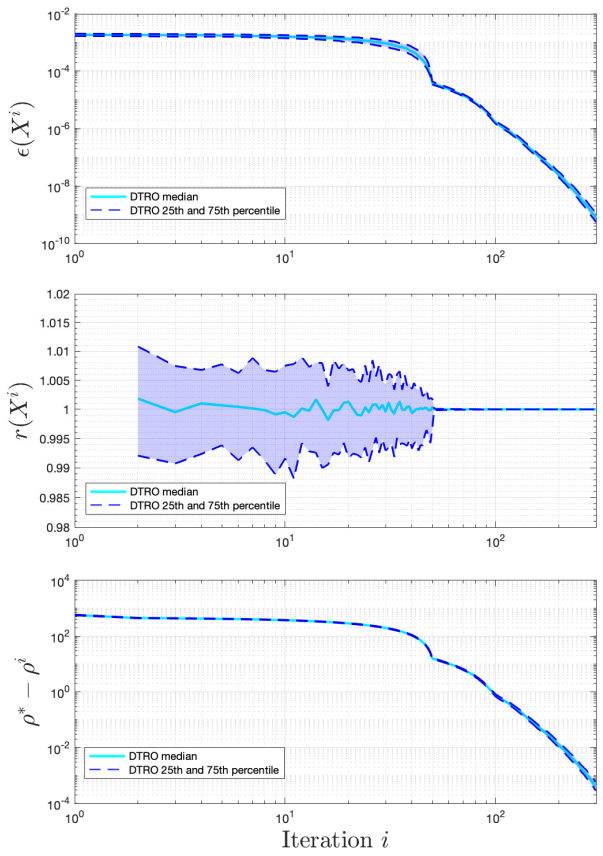


Fig. 1: Convergence of DTRO. *Top*: MSE of the entries of $X^i$ compared to $X^*$. *Center*: Estimation of the convergence rate. *Bottom*: Convergence in objective.

consecutive objectives. For the DTRO algorithm (Algorithm 2), we fixed the number of iterations to 300 i.e., six full update rounds. In Fig. 1 (Top), the $\epsilon$ function corresponds to the Mean Squared Error (MSE) between the solutions of both algorithms:

$$\epsilon(X^i) = \frac{1}{MQ}||X^i - X^*||_F^2. \tag{23}$$

We estimate the convergence rate of the DTRO algorithm by analysing the function $r$ plotted in Fig. 1 (Center):

$$r(X^i) = \frac{||X^i - X^*||_F}{||X^{i-1} - X^*||_F}. \tag{24}$$

If a sequence $\{X^i\}_i$ satisfies $\lim_{i\to+\infty} r(X^i) = 1$, the sequence is said to converge sublinearly to $X^*$. In the case of the DTRO algorithm, $r$ approaches 1 as the number of iteration grows, therefore it is estimated that the DTRO algorithm has a convergence rate close to sublinear in this simulation scenario, as shown in Fig. 1 (Bottom).

These results allow us to visualize the claimed convergence of the sequence $\{X^i\}_i$ to the optimum $X^*$. In particular, we can observe abrupt changes in certain plots of the DTRO algorithm at the end of the first full round update i.e., $i = 50$. Due to the constraints on a given updating node $q$ at iteration $i$ on the freedom of choosing a new $X_k^{i+1} = X_k^i G_k$ for other nodes $k \neq q$, we cannot expect to have a reliable estimate before the first round if we set $X^0$ randomly.

## V. Conclusions and Future Works

We have proposed a distributed algorithm to solve the TRO problem given in (2). By partially solving a network-wide EVD by the means of a local GEVD problem at each iteration, where only compressed versions of signals measured throughout the network are communicated, we achieved convergence at a rate around sublinear according to our simulations. Adapting this algorithm to partially connected topologies can be considered as an interesting future direction of study, along with analysing the effect of asynchronous updates in the network.

## References


[1] Y.-F. Guo, S.-J. Li, J.-Y. Yang, T.-T. Shu, and L.-D. Wu, "A generalized Foley–Sammon transform based on generalized Fisher discriminant criterion and its application to face recognition," *Pattern Recognition Letters*, vol. 24, no. 1-3, pp. 147–158, 2003.

[2] H. Wang, S. Yan, D. Xu, X. Tang, and T. Huang, "Trace ratio vs. ratio trace for dimensionality reduction," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.

[3] Y. Jia, F. Nie, and C. Zhang, "Trace ratio problem revisited," *IEEE Transactions on Neural Networks*, vol. 20, no. 4, pp. 729–735, 2009.

[4] S. Yan and X. Tang, "Trace quotient problems revisited," in *European Conference on Computer Vision*. Springer, 2006, pp. 232–244.

[5] F. Nie, S. Xiang, Y. Jia, C. Zhang, and S. Yan, "Trace ratio criterion for feature selection." in *AAAI*, vol. 2, 2008, pp. 671–676.

[6] C. Shen, H. Li, and M. J. Brooks, "Supervised dimensionality reduction via sequential semidefinite programming," *Pattern Recognition*, vol. 41, no. 12, pp. 3644–3652, 2008.

[7] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[8] J. W. Sammon, "An optimal discriminant plane," *IEEE Transactions on Computers*, vol. 100, no. 9, pp. 826–829, 1970.

[9] D. H. Foley and J. W. Sammon, "An optimal set of discriminant vectors," *IEEE Transactions on Computers*, vol. 100, no. 3, pp. 281–289, 1975.

[10] K. Liu, Y.-Q. Cheng, J.-Y. Yang, and X. Liu, "An efficient algorithm for Foley–Sammon optimal set of discriminant vectors by algebraic method," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 6, no. 05, pp. 817–829, 1992.

[11] K. Liu, Y.-Q. Cheng, and J.-Y. Yang, "A generalized optimal set of discriminant vectors," *Pattern Recognition*, vol. 25, no. 7, pp. 731–739, 1992.

[12] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[13] J. Asensio-Cubero, J. Q. Gan, and R. Palaniappan, "Extracting optimal tempo-spatial features using local discriminant bases and common spatial patterns for brain computer interfacing," *Biomedical Signal Processing and Control*, vol. 8, no. 6, pp. 772–778, 2013.

[14] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Common spatial pattern revisited by Riemannian geometry," in *2010 IEEE International Workshop on Multimedia Signal Processing*. IEEE, 2010, pp. 472–476.

[15] A. Bertrand, "Distributed signal processing for wireless EEG sensor networks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 6, pp. 923–935, 2015.

[16] A. M. Narayanan and A. Bertrand, "Analysis of miniaturization effects and channel selection strategies for EEG sensor networks with application to auditory attention detection," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 1, pp. 234–244, 2020.

[17] Y. Wang, S. Gao, and X. Gao, "Common spatial pattern method for channel selelction in motor imagery based brain-computer interface," in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*. IEEE, 2006, pp. 5392–5395.

[18] A. Bertrand and M. Moonen, "Distributed adaptive generalized eigenvector estimation of a sensor signal covariance matrix pair in a fully connected sensor network," *Signal Processing*, vol. 106, pp. 209–214, 2015.

[19] J. Szurley, A. Bertrand, and M. Moonen, "Distributed adaptive node-specific signal estimation in heterogeneous and mixed-topology wireless sensor networks," *Signal Processing*, vol. 117, pp. 44–60, 2015.

[20] J. Ye, R. Janardan, and Q. Li, "Two-dimensional linear discriminant analysis," in *Advances in neural information processing systems*, 2005, pp. 1569–1576.

[21] S. Xiang, F. Nie, and C. Zhang, "Learning a Mahalanobis distance metric for data clustering and classification," *Pattern Recognition*, vol. 41, no. 12, pp. 3600–3612, 2008.

[22] D. Cai, X. He, J. Han, and H.-J. Zhang, "Orthogonal Laplacianfaces for face recognition," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3608–3614, 2006.

[23] T. T. Ngo, M. Bellalij, and Y. Saad, "The trace ratio optimization problem," *SIAM review*, vol. 54, no. 3, pp. 545–569, 2012.

[24] A. Bertrand and M. Moonen, "Distributed adaptive estimation of covariance matrix eigenvectors in wireless sensor networks with application to distributed PCA," *Signal Processing*, vol. 104, pp. 120–135, 2014.