Low-complexity distributed total least squares estimation in ad hoc sensor networks

Alexander Bertrand*†, *Member, IEEE,* and Marc Moonen*†, *Fellow, IEEE,*

∗ KU Leuven, Dept. of Electrical Engineering-ESAT, SCD-SISTA
† IBBT Future Health Department
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
E-mail: alexander.bertrand@esat.kuleuven.be
marc.moonen@esat.kuleuven.be
Phone: +32 16 321899, Fax: +32 16 321970

***Abstract*—Total least squares (TLS) estimation is a popular solution technique for overdetermined systems of linear equations with a noisy data matrix. In this paper, we revisit the distributed total least squares (D-TLS) algorithm, which operates in an ad hoc network, where each node has access to a subset of the linear equations. The D-TLS algorithm computes the TLS solution of the full system of equations in a fully distributed fashion (without fusion center). To reduce the large computational complexity due to an eigenvalue decomposition (EVD) at every node and in each iteration, we modify the D-TLS algorithm based on inverse power iterations (IPIs). In each step of the modified algorithm, a single IPI is performed, which significantly reduces the computational complexity. We show that this IPI-based D-TLS algorithm still converges to the network-wide TLS solution under certain assumptions, which are often satisfied in practice. We provide simulation results to demonstrate the convergence of the algorithm, even when some of these assumptions are not satisfied.**

EDICS: SEN-DIST Distributed signal processing

***Index Terms*—Distributed estimation, wireless sensor networks (WSNs), total least squares, inverse power iteration**

I. Introduction

In this paper, we consider the linear regression problem $\mathbf{U}\mathbf{w} = \mathbf{d}$ in the unknown P-dimensional regressor $\mathbf{w}$, with $\mathbf{U}$ an $M \times P$ regression matrix and $\mathbf{d}$ an M-dimensional data vector with $M \geq P$. Total least squares (TLS) estimation is a popular solution method for this type of problems, when both the regression matrix $\mathbf{U}$ and the data vector $\mathbf{d}$ are corrupted by additive noise [2]. The TLS solution is computed from the eigenvector corresponding to the smallest eigenvalue of the (squared) extended data matrix $[\mathbf{U} \mid \mathbf{d}]^T [\mathbf{U} \mid \mathbf{d}]$, and therefore relies on an eigenvalue decomposition (EVD).

The problem of solving an overdetermined set of equations is often encountered in sensor network applications, where nodes can either have access to subsets of the columns of $\mathbf{U}$, e.g. for distributed signal enhancement and beamforming [3]–[5], or to subsets of the equations, i.e., subsets of the rows of $\mathbf{U}$ and $\mathbf{d}$ [6]–[12], e.g. for distributed system identification. The challenge is then to solve the network-wide problem in a distributed fashion, i.e., without gathering all the data in a fusion center.

In [13], the distributed TLS problem is defined in a wireless sensor network (WSN), where the rows of $\mathbf{U}$ and their corresponding entries in $\mathbf{d}$ are distributed over the nodes. A related distributed TLS problem has later been applied in [14] for cognitive spectrum sensing. In both [13] and [14], the goal is to find the solution of the full system of equations in a distributed fashion, i.e., without gathering all the equations in a central processing unit. Due to the non-convex nature of the problem, the proposed algorithm in [14] computes a stationary point of the cost function, based on a combination of a block coordinate descent method and the alternating direction method of multipliers, which eventually yields nested subiterations. The distributed TLS (D-TLS) algorithm proposed in [13] avoids such nested subiteration by applying a convex relaxation. Despite this relaxation, it has been proven that this D-TLS algorithm converges to the exact TLS solution in each node. In each iteration of the D-TLS algorithm, each node solves a local TLS problem and shares its result with its neighbors. The nodes then combine the received data from their neighbors to update their local TLS matrix. Due to the flexibility and the uniformity of the nodes in the network, there is no single point of failure, which makes the algorithm robust to sensor failures.

Each node in the D-TLS algorithm solves a local TLS problem in each iteration, which requires an EVD. This is a significant computational task if the unknown regressor $\mathbf{w}$ has a large dimension. Fortunately, since the D-TLS algorithm

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

The work of A. Bertrand was supported by a Postdoctoral Fellowship of the Research Foundation - Flanders (FWO). This work was carried out at the ESAT Laboratory of KU Leuven, in the frame of KU Leuven Research Council CoE EF/05/006 'Optimization in Engineering' (OPTEC) and PFV/10/002 (OPTEC), Concerted Research Action GOA-MaNet, the Belgian Programme on Interuniversity Attraction Poles initiated by the Belgian Federal Science Policy Office IUAP P6/04 (DYSCO, 'Dynamical systems, control and optimization', 2007-2011), Research Project IBBT, and Research Project FWO nr. G.0763.12 ('Wireless acoustic sensor networks for extended auditory communication'). The scientific responsibility is assumed by its authors. A conference precursor of this manuscript has been submitted as [1].

only requires the eigenvector corresponding to the smallest eigenvalue of the local extended data matrix, the EVD can be replaced by an inverse power iteration (IPI) procedure. However, to find the corresponding eigenvector, multiple inverse power (IP) subiterations are required for each iteration of the D-TLS algorithm, which again represents a significant computational burden.

In this paper, we relax the task for each node to exactly solve a TLS problem in each iteration. Instead, each node performs a *single* IP subiteration, and then shares the resulting vector with its neighbors. This significantly decreases the computational burden at each node. Furthermore, this results in a smooth transition of the local regressor estimates, making it easier for the algorithm to reach a consensus. This is especially important in situations where the distance between the smallest and one but smallest eigenvalue of the (squared) local extended data matrix is small. In the original D-TLS algorithm, the local regressor estimates can then change very abruptly due to recurring 'swaps' of the eigenvectors in successive iterations. We refer to the new algorithm as the IPI-D-TLS algorithm. Since the local TLS problems are only approximately solved in each iteration of the IPI-D-TLS algorithm, the theoretical convergence results from [13] are no longer valid. Nevertheless, we prove that under certain conditions, which are often satisfied in practice, the IPI-D-TLS algorithm converges to the network-wide D-TLS solution at each node. Most of these conditions were also required to prove convergence of the original D-TLS algorithm. We demonstrate by means of simulations that convergence is still obtained, even if some of the conditions are not met (e.g. when using a fixed step size instead of a decreasing step size).

It is noted that D-TLS and IPI-D-TLS are actually extreme cases of a family of algorithms, each having a different number of IP subiterations. Indeed, D-TLS (virtually) applies an infinite amount of IP subiterations, and IPI-D-TLS applies a single IP subiteration. This family of IPI-based D-TLS algorithms yields a tradeoff between computational complexity and communication load (in batch mode) or tracking performance (in adaptive mode), since reducing the number of IP subiterations requires more high-level iterations/data exchange to obtain the same accuracy. In this paper, we only focus on the convergence properties of IPI-D-TLS (with a single IP subiteration). It is reasonable to assume that convergence of this algorithm implies convergence of all other IPI-based D-TLS algorithms with multiple IP subiterations (under the same conditions).

The paper is organized as follows. In Section II, we briefly review the distributed TLS problem statement and the D-TLS algorithm, as well as some of its convergence properties. The D-TLS algorithm is then modified to the IPI-D-TLS algorithm in Section III, and its convergence properties are addressed in Section IV. Section V contains the main part of this paper, i.e., the convergence proof of the IPI-D-TLS algorithm. In Section VI, we provide simulation results. Conclusions are drawn in Section VII.

II. Distributed Total Least Squares (D-TLS) algorithm

In this section, we briefly review the distributed TLS problem statement and the D-TLS algorithm. For further details, we refer to [13].

Consider an ad hoc WSN with the set of nodes $\mathcal{K} = \{1, \ldots, K\}$ and with a random (connected) topology, where nodes can exchange data with their neighbors through a wireless link. We denote the set of neighbor nodes of node k as $\mathcal{N}_k$, i.e., all the nodes that can share data with node k, node k excluded. $|\mathcal{N}_k|$ denotes cardinality of the set $\mathcal{N}_k$, i.e., the number of neighbors of node k. Node k collects a noisy $M_k \times P$ data matrix $\mathbf{U}_k = \widetilde{\mathbf{U}}_k + \mathbf{N}_k$ and a noisy M_k-dimensional data vector $\mathbf{d}_k = \tilde{\mathbf{d}}_k + \mathbf{n}_k$, for which the clean versions $\widetilde{\mathbf{U}}_k$ and $\tilde{\mathbf{d}}_k$ are assumed to be related through a linear regressor $\mathbf{w}$, i.e., $\widetilde{\mathbf{U}}_k \mathbf{w} = \tilde{\mathbf{d}}_k$. The goal is then to solve a TLS problem for the network-wide system of equations in which all $\mathbf{U}_k$'s and $\mathbf{d}_k$'s are stacked, i.e. to compute the regressor vector $\mathbf{w}$ from

$$\min_{\mathbf{w}, \triangle\mathbf{U}_1, \ldots, \triangle\mathbf{U}_K, \triangle\mathbf{d}_1, \ldots, \triangle\mathbf{d}_K} \sum_{k \in \mathcal{K}} \left(\|\triangle\mathbf{U}_k\|_F^2 + \|\triangle\mathbf{d}_k\|^2 \right) \quad (1)$$

$$\text{s.t. } (\mathbf{U}_k + \triangle\mathbf{U}_k)\,\mathbf{w} = (\mathbf{d}_k + \triangle\mathbf{d}_k)\,,\ k = 1, \ldots, K \quad (2)$$

where $\|.\|_F$ and $\|.\|$ denote the Frobenius norm and the 2-norm, respectively. In [15], it is shown that the TLS estimate is unbiased when the noises $\mathbf{N}_k$ and $\mathbf{n}_k$ contaminating $\widetilde{\mathbf{U}}_k$ and $\tilde{\mathbf{d}}_k$ are zero mean and uncorrelated over all entries of the $\mathbf{U}_k$'s and $\mathbf{d}_k$'s.

Problem (1)-(2) is referred to as the distributed TLS problem, since each node only has access to a part of the data. Its solution for $\mathbf{w}$ is denoted as $\mathbf{w}^*$. The goal is to compute $\mathbf{w}^*$ in a distributed fashion, without gathering all data in a fusion center.

In the sequel, $\mathbf{e}_N$ is an N-dimensional vector with all zeros except for the last entry which is equal to 1, $\mathbf{0}_N$ is the N-dimensional all-zero vector, $\mathbf{I}_N$ denotes the $N \times N$ identity matrix, and $\mathbf{O}_{N \times Q}$ is the $N \times Q$ all-zero matrix. Let $N = P + 1$ and define the $N \times N$ matrix $\mathbf{R}_k = \mathbf{U}_{k+}^T \mathbf{U}_{k+}$, with $\mathbf{U}_{k+} = [\mathbf{U}_k \mid \mathbf{d}_k]$. Then the solution of the distributed TLS problem is given by [2]:

$$\mathbf{w}^* = -\frac{1}{\mathbf{e}_N^T \mathbf{x}^*} \left[\begin{array}{c|c} \mathbf{I}_P & \mathbf{0}_P \end{array} \right] \mathbf{x}^* \quad (3)$$

where $\mathbf{x}^*$ is the eigenvector corresponding to the smallest eigenvalue of $\mathbf{R} = \sum_{k \in \mathcal{K}} \mathbf{R}_k$. If the smallest eigenvalue degenerates, i.e., if it has multiplicity larger than one, the TLS solution is non-unique, and then it may be desirable to single out a minimum-norm solution [2]. Furthermore, in rare and contrived scenarios, it may occur that $\mathbf{e}_N^T \mathbf{x}^* = 0$, such that (3) cannot be used. This is known in literature as a non-generic TLS problem, which has to be solved in a different way [2]. Both these special cases are beyond the scope of this paper, hence we assume that $\mathbf{x}^*$ is unique, and $\mathbf{e}_N^T \mathbf{x}^* \neq 0$.

The eigenvector $\mathbf{x}^*$ can be computed in an iterative distributed fashion by means of the D-TLS algorithm [13], which is given in Table I. In the algorithm description in Table I, a new matrix $\mathbf{\Theta}^{(i)}$ has been introduced, which contains the

TABLE I
THE DISTRIBUTED TOTAL LEAST SQUARES (D-TLS) ALGORITHM.

D-TLS Algorithm

1) $\forall\, k \in \mathcal{K}$: Initialize $\mathbf{\Theta}_k^{(0)} = \mathbf{O}_{N\times N}$, with $N = P+1$.
2) $i \leftarrow 0$
3) Each node $k \in \mathcal{K}$ computes the eigenvector $\mathbf{x}_k^{(i)}$ corresponding to the smallest eigenvalue of $\overline{\mathbf{R}}_k^{(i)}$ defined by
$$\overline{\mathbf{R}}_k^{(i)} = \mathbf{R}_k + \mathbf{\Theta}_k^{(i)} \tag{4}$$
where $\mathbf{x}_k^{(i)}$ is scaled such that $\|\mathbf{x}_k^{(i)}\| = 1$.
4) Each node $k \in \mathcal{K}$ transmits $\mathbf{x}_k^{(i)}$ to the nodes in $\mathcal{N}_k$.
5) Each node $k \in \mathcal{K}$ updates
$$\mathbf{\Theta}_k^{(i+1)} = \mathbf{\Theta}_k^{(i)} + \mu_i \left(|\mathcal{N}_k| \mathbf{x}_k^{(i)} \mathbf{x}_k^{(i)\,T} - \sum_{q\in\mathcal{N}_k} \mathbf{x}_q^{(i)} \mathbf{x}_q^{(i)\,T} \right) \tag{5}$$
with stepsize $\mu_i > 0$.
6) Compute the local TLS solution $\mathbf{w}_k^{(i)} = -\frac{1}{\mathbf{e}_N^T \mathbf{x}_k^{(i)}} \left[\begin{array}{c|c} \mathbf{I}_P & \mathbf{0}_P \end{array}\right] \mathbf{x}_k^{(i)}$.
7) $i \leftarrow i+1$.
8) return to step 3.

Lagrange multipliers corresponding to some matrix constraints that enforce consensus between nodes. The update (5) is a subgradient update on these Lagrange multipliers, based on a dual decomposition of the original optimization problem (for further details, we refer to [13]).

In [13], it has been shown that, $\forall\, k \in \mathcal{K}$, the $\mathbf{w}_k^{(i)}$ in the D-TLS algorithm converges to (3) under the step size conditions

$$\sum_{i=0}^{\infty} \mu_i = \infty \tag{6}$$

$$\sum_{i=0}^{\infty} (\mu_i)^2 < \infty\,. \tag{7}$$

Under the same conditions, $\mathbf{x}_k^{(i)}$ converges to the eigenvector corresponding to the smallest eigenvalue of $\mathbf{R} = \sum_{k\in\mathcal{K}} \mathbf{R}_k$. This is also the eigenvector corresponding to the smallest eigenvalue of $\lim_{i\to\infty} \overline{\mathbf{R}}^{(i)}$ where $\overline{\mathbf{R}}^{(i)} = \sum_{k\in\mathcal{K}} \overline{\mathbf{R}}_k^{(i)}$ since $\sum_{k\in\mathcal{K}} \mathbf{\Theta}_k^{(i)} = \mathbf{O}_{N\times N}$, $\forall\, i \in \mathbb{N}$, and therefore $\overline{\mathbf{R}}^{(i)} = \mathbf{R}$, $\forall\, i \in \mathbb{N}$.

Remark I: Table I shows the second version of the D-TLS algorithm (see [13], Section IV-B, Remark I), which is theoretically equivalent to the original version, but which has a reduced overhead and memory usage at each node. Although this simplified version is robust against random temporary link failures, it cannot heal itself from permanent link failures [13]. Furthermore, it is quite sensitive to communication noise in the wireless links, hence requiring strong channel codes. Indeed, notice that noise on the transmitted $\mathbf{x}_k$'s will accumulate in the $\mathbf{\Theta}_k$'s in (5) such that their sum will deviate from zero ($\sum_{k\in\mathcal{K}} \mathbf{\Theta}_k^{(i)} \neq \mathbf{O}_{N\times N}$), which has a significant impact on the algorithm. In the original version of the D-TLS algorithm, this problem can be circumvented by letting neighboring nodes communicate their shared Lagrange multipliers every now and then to make them consistent at both nodes (note that this increases the required communication bandwidth). It is important to note that all results in this paper also apply to the original version of D-TLS, i.e., the IPI-D-TLS algorithm as derived in Section III can be straightforwardly reformulated in this alternative version, to inherit these robustness properties. However, for the sake of an easy exposition, we only consider the simplified version based on the algorithm description in Table I, and we refer to [13] for more details on the alternative implementation.

III. D-TLS WITH INVERSE POWER ITERATIONS (IPI-D-TLS)

The computation of the eigenvector corresponding to the smallest eigenvalue of $\overline{\mathbf{R}}_k^{(i)}$ is the most expensive step of the D-TLS algorithm, which has to be performed in all nodes $k \in \mathcal{K}$ and in all iterations $i \in \mathbb{N}$. In this section, we will modify the D-TLS algorithm such that this step is replaced by a single $N \times N$ matrix-vector multiplication.

Let $\overline{\mathbf{R}}_k^{(i)} = \mathbf{B}_k^{(i)} \mathbf{\Lambda}_k^{(i)} \mathbf{B}_k^{(i)\,T}$ denote the eigenvalue decomposition of $\overline{\mathbf{R}}_k^{(i)}$ where $\mathbf{\Lambda}_k^{(i)} = \mathrm{diag}\{\lambda_{k,1}^{(i)}, \ldots, \lambda_{k,N}^{(i)}\}$ such that $\lambda_{k,1}^{(i)} \geq \lambda_{k,2}^{(i)} \geq \ldots \geq \lambda_{k,N}^{(i)}$, $\mathbf{B}_k^{(i)}$ is an orthogonal matrix, and $\mathbf{b}_{k,j}^{(i)}$ is its j-th column. Assume $\lambda_{k,N-1}^{(i)} > \lambda_{k,N}^{(i)} > 0$ (this is motivated later), and let

$$\overline{\mathbf{P}}_k^{(i)} = \left(\overline{\mathbf{R}}_k^{(i)}\right)^{-1} \tag{8}$$

then it is well-known that the following iterative procedure, referred to as the inverse power iteration (IPI) method, converges to the eigenvector $\mathbf{b}_{k,N}^{(i)}$:

1) Initialize the N-dimensional vector $\mathbf{x}$ with a unit-norm vector such that $\mathbf{x}^T \mathbf{b}_{k,N}^{(i)} \neq 0$.

2) Repeat until convergence:

$$\mathbf{x} \leftarrow \frac{\overline{\mathbf{P}}_k^{(i)}\mathbf{x}}{\|\overline{\mathbf{P}}_k^{(i)}\mathbf{x}\|} \ . \tag{9}$$

Therefore, we could replace step 3 in the D-TLS algorithm by the above IPI procedure. Assuming that the stepsize μ_i is not too large, the eigenvectors $\mathbf{b}_{k,N}^{(i-1)}$ of $\overline{\mathbf{R}}_k^{(i-1)}$ and $\mathbf{b}_{k,N}^{(i)}$ of $\overline{\mathbf{R}}_k^{(i)}$ will be close to each other, and hence only a small number of IPI's is required in the i-th iteration if the IPI method is initialized with the computed eigenvector $\mathbf{x}_k^{(i-1)}$ from the previous iteration. In the modified algorithm, we will therefore only perform a single IPI in each D-TLS iteration.

It is noted that the above procedure requires the inversion of $\overline{\mathbf{R}}_k^{(i)}$ to obtain $\overline{\mathbf{P}}_k^{(i)}$, which is an $O(N^3)$ procedure. A recursive update of $\overline{\mathbf{P}}_k^{(i-1)}$ to obtain $\overline{\mathbf{P}}_k^{(i)}$ significantly reduces the computational complexity, assuming that $N \gg |\mathcal{N}_k|$. Indeed, (5) combined with (4) shows that $\overline{\mathbf{R}}_k^{(i)}$ is derived from $\overline{\mathbf{R}}_k^{(i-1)}$ by means of $|\mathcal{N}_k|+1$ rank-1 updates. With the Woodbury matrix identity [16, Section 2.1.3], the inverse of a symmetric rank-1 updated matrix $\mathbf{A}$ can be efficiently computed as

$$\left(\mathbf{A} \pm \mathbf{x}\mathbf{x}^T\right)^{-1} = \mathbf{A}^{-1} \mp \frac{\left(\mathbf{A}^{-1}\mathbf{x}\right)\left(\mathbf{x}^T\mathbf{A}^{-1}\right)}{1 \pm \mathbf{x}^T\mathbf{A}^{-1}\mathbf{x}} \ . \tag{10}$$

Assuming $\overline{\mathbf{P}}_k^{(i-1)}$ is known, we can compute $\overline{\mathbf{P}}_k^{(i)}$ by applying this rank-1 update $|\mathcal{N}_k|+1$ times, which is an $O(|\mathcal{N}_k|N^2)$ procedure. For brevity, we define $R1^{\pm}\left(\mathbf{A}^{-1},\mathbf{x}\right)$ as the operator that computes $\left(\mathbf{A} \pm \mathbf{x}\mathbf{x}^T\right)^{-1}$ from $\mathbf{A}^{-1}$ and $\mathbf{x}$ based on (10), i.e.,

$$R1^{\pm}\left(\mathbf{A}^{-1},\mathbf{x}\right) = \mathbf{A}^{-1} \mp \frac{\left(\mathbf{A}^{-1}\mathbf{x}\right)\left(\mathbf{x}^T\mathbf{A}^{-1}\right)}{1 \pm \mathbf{x}^T\mathbf{A}^{-1}\mathbf{x}} \ . \tag{11}$$

This in effect yields the IPI-D-TLS algorithm, as described in Table II. It is noted that the cooperation among the nodes occurs in step 5 of the IPI-D-TLS algorithm, which is actually equivalent to the combination of (4) and (5) in the original D-TLS algorithm. However, the update is here explicitely applied on the inverse matrix $\overline{\mathbf{P}}_k^{(i)} = \left(\overline{\mathbf{R}}_k^{(i)}\right)^{-1}$. In fact, this step implements an $O(|\mathcal{N}_k|N^2)$ procedure to compute

$$\left(\overline{\mathbf{R}}_k^{(i+1)}\right)^{-1} = \left(\overline{\mathbf{R}}_k^{(i)} + \mu_i\left(|\mathcal{N}_k|\mathbf{x}_k^{(i)}\mathbf{x}_k^{(i)\,T} - \sum_{q\in\mathcal{N}_k}\mathbf{x}_q^{(i)}\mathbf{x}_q^{(i)\,T}\right)\right)^{-1} . \tag{12}$$

This step also dominates the computations, hence the overall complexity at node k is $O(|\mathcal{N}_k|N^2)$ per iteration.

Remark II: A rough complexity analysis yields the following required number of floating point operations (flops) per node and per iteration:

- D-TLS: $21N^3 + (3|\mathcal{N}_k|+4)N^2 + (|\mathcal{N}_k|+1)N$
- IPI-D-TLS: $(5|\mathcal{N}_k|+2)N^2 + (4|\mathcal{N}_k|+2)N$

Here, we assume that an eigenvalue decomposition (or a singular value decomposition) of an $N \times N$ matrix requires $21N^3$ flops [16], and a matrix multiplication of an $M \times N$ matrix with an $N \times Q$ matrix requires $2MNQ$ flops. Even for a TLS problem of small dimension, e.g., $N = 10$, one iteration of the IPI-D-TLS algorithm requires only 8% of the flops that are required for the D-TLS algorithm (for a node with 3 neighbors). This difference rapidly becomes even more significant if N grows, e.g., if $N = 100$, one iteration of the IPI-D-TLS algorithm requires only 0.8% of the number of flops required for one iteration of the D-TLS algorithm.

IV. Convergence properties of the IPI-D-TLS algorithm

We will prove convergence of the IPI-D-TLS algorithm under the following assumptions:

Assumption 1: $\sum_{i=0}^{\infty}\mu_i = \infty$

Assumption 2: $\sum_{i=0}^{\infty}(\mu_i)^2 < \infty$

Assumption 3: $\exists\,\kappa_1 > 0,\ \exists\,L_1 \in \mathbb{N},\ \forall\,k \in \mathcal{K}:\ i > L_1 \Rightarrow \lambda_{k,N-1}^{(i)} - \lambda_{k,N}^{(i)} > \kappa_1$

Assumption 4: $\exists\,\xi_1 > 0, \forall\,i \in \mathbb{N},\ \forall\,k \in \mathcal{K}:\ \lambda_{k,N}^{(i)} > \xi_1$

It is noted that Assumptions 1, 2 and 3 are also required to guarantee convergence of the original D-TLS algorithm.

The following theorem guarantees the convergence and optimality of the IPI-D-TLS algorithm under these assumptions:

Theorem IV.1. *Let Assumptions 1 to 4 be satisfied. Then the following holds for the IPI-D-TLS algorithm:*

$$\forall\,k \in \mathcal{K}:\ \lim_{i\to\infty}\mathbf{w}_k^{(i)} = \mathbf{w}^* \tag{14}$$

where $\mathbf{w}^*$ *is the solution of (1)-(2), given in (3).*

The proof of this theorem is elaborate and can be found in Section V. However, we will first discuss the assumptions that were made, and how they impact the implementation and behavior of the IPI–D-TLS algorithm in practice.

A. Assumptions 1 and 2

Assumptions 1 and 2 are often imposed in convergence proofs of (sub)gradient, stochastic gradient or relaxation methods (see e.g. [13], [17]–[20]), and indeed already appeared in expressions (6)-(7) to guarantee convergence of the D-TLS algorithm. As with the D-TLS algorithm, Assumption 2 may yield slow convergence in the IPI-D-TLS algorithm, and it is not a practical assumption in tracking applications. However, the fact that convergence can be proven under these conditions is good news since it means that in principle an infinite accuracy can be obtained. Furthermore, this usually indicates that the algorithm will at least converge to a neighborhood of the exact solution, when using a fixed step size that is sufficiently small. This neighborhood then shrinks with the chosen step size. Simulations in Section VI will demonstrate that this is indeed also the case for the IPI-D-TLS algorithm.

B. Assumption 3

Assumption 3 guarantees that, after sufficiently many iterations, the smallest and one but smallest eigenvalue of the matrix $\overline{\mathbf{R}}_k^{(i)}$ are well-separated in each node, i.e., the

TABLE II
THE INVERSE POWER ITERATION D-TLS ALGORITHM.

IPI-D-TLS Algorithm

1) $\forall\, k \in \mathcal{K}$: Initialize $\overline{\mathbf{P}}_k^{(1)} = (\mathbf{R}_k)^{-1}$ and choose a unit-norm N-dimensional vector $\mathbf{x}_k^{(0)}$.
2) $i \leftarrow 1$
3) Each node $k \in \mathcal{K}$ computes
$$\mathbf{x}_k^{(i)} = \frac{\overline{\mathbf{P}}_k^{(i)} \mathbf{x}_k^{(i-1)}}{\|\overline{\mathbf{P}}_k^{(i)} \mathbf{x}_k^{(i-1)}\|} \,. \tag{13}$$
4) Each node $k \in \mathcal{K}$ transmits $\mathbf{x}_k^{(i)}$ to the nodes in $\mathcal{N}_k$.
5) Each node $k \in \mathcal{K}$ computes $\overline{\mathbf{P}}_k^{(i+1)}$ as follows (for stepsize $\mu_i > 0$)
   - $\mathbf{A} \leftarrow \overline{\mathbf{P}}_k^{(i)}$
   - $\forall\, q \in \mathcal{N}_k :\ \mathbf{A} \leftarrow R1^{-}\left(\mathbf{A}, \sqrt{\mu_i}\mathbf{x}_q^{(i)}\right)$
   - $\overline{\mathbf{P}}_k^{(i+1)} = R1^{+}\left(\mathbf{A}, \sqrt{\mu_i |\mathcal{N}_k|}\mathbf{x}_k^{(i)}\right)$.
6) Compute the local TLS solution $\mathbf{w}_k^{(i)} = -\frac{1}{\mathbf{e}_N^T \mathbf{x}_k^{(i)}} \left[\ \mathbf{I}_P \ \middle|\ \mathbf{0}_P\ \right] \mathbf{x}_k^{(i)}$.
7) $i \leftarrow i + 1$.
8) return to step 3.

smallest eigenvalue does not degenerate[1]. It is noted that this eigenvalue degeneration is not a problem as such, as long as the eigenvalue later iterates away from this degeneration, which is usually the case. Only if the smallest eigenvalue degenerates in (or near) the accumulation point $\overline{\mathbf{R}}_k^{(\infty)}$, a problem may occur to reach consensus in the $\mathbf{w}_k^{(i)}$'s. This certainly holds for the D-TLS algorithm, since step 3) becomes ill-defined in such situations, i.e., when $\lambda_{k,N}^{(i)} \approx \lambda_{k,N-1}^{(i)}$, small changes in the eigenvalues may result in a completely different $\mathbf{x}_k^{(i)}$ (note that this is not the case for IPI-D-TLS which is more effective in such situations). This swap of eigenvectors can only be resolved by choosing a smaller μ_i, or by adding extra functionality to the algorithm to ensure that each node selects the same eigenvector from the eigenspace spanned by $\{\mathbf{b}_{k,N}^{(i)}, \mathbf{b}_{k,N-1}^{(i)}\}$. We will not further elaborate on the case of degenerated eigenvalues, since it rarely occurs and it significantly complicates the theoretical results. However, it is an interesting fundamental problem that requires further investigation, which is beyond the scope of this paper.

C. Assumption 4

Assumption 4 guarantees that the smallest eigenvalue remains strictly positive. This avoids rank deficiency of $\overline{\mathbf{R}}_k^{(i)}$, and it avoids that the IPI method converges to an eigenvector other than $\mathbf{b}_{k,N}^{(i)}$ whenever the corresponding eigenvalue $\lambda_{k,N}^{(i)} < 0$ and $\exists\ j < N :\ |\lambda_{k,j}^{(i)}| < |\lambda_{k,N}^{(i)}|$. Assumption 4 is a reasonable assumption for small step sizes μ_i and sufficiently large initial eigenvalues $\lambda_{k,N}^{(0)}$'s. If the smallest eigenvalue does become negative, one can always add a scaled identity matrix to $\overline{\mathbf{R}}_k^{(i)}$ (and make the corresponding changes in $\overline{\mathbf{P}}_k^{(i)} = \left(\overline{\mathbf{R}}_k^{(i)}\right)^{-1}$). This does not affect the solution of the D-TLS problem, since the sum $\sum_{k\in\mathcal{K}} \overline{\mathbf{R}}_k^{(i)}$ is then equal to $\nu\mathbf{I}_N + \sum_{k\in\mathcal{K}} \mathbf{R}_k = \nu\mathbf{I}_N + \mathbf{R}$ (for some ν), which indeed has the same eigenvectors and the same ranking of eigenvalues as $\mathbf{R}$.

[1]In the sequel, we say that the smallest eigenvalue $\lambda_{k,N}^{(i)}$ degenerates if the distance $|\lambda_{k,N}^{(i)} - \lambda_{k,N-1}^{(i)}| < 2|\mathcal{N}_k|\mu_i$, since only then the update (5) may result in a 'swap' of eigenvectors.

V. CONVERGENCE PROOF

This section is devoted to the proof of Theorem IV.1. However, we first need to introduce some preliminary definitions and theorems that will be needed in the sequel.

A. Preliminaries

We will need the following lemma, which is proven in Appendix A:

Lemma V.1. *If Assumptions 3 and 4 are satisfied, then*

$$\exists\, \xi_2 > 0, \forall\, i \in \mathbb{N},\ \forall\, k \in \mathcal{K} :\ \lambda_{k,1}^{(i)} < \xi_2 \tag{15}$$

and

$$\exists\, 0 < \beta < 1,\ \exists\, L_1 \in \mathbb{N},\ \forall\, k \in \mathcal{K} :\ i > L_1 \Rightarrow \frac{\lambda_{k,N}^{(i)}}{\lambda_{k,N-1}^{(i)}} < \beta \,. \tag{16}$$

Furthermore, we also need some definitions and theorems from the convergence analysis of the D-TLS algorithm (details can be found in [13]). First, it is noted that (5) in Table I defines a subgradient update which results in a maximization of a concave Lagrange dual function denoted by $d(\mathbf{\Theta})$, with $\mathbf{\Theta}$ denoting the $KN \times N$ matrix that stacks all the $\mathbf{\Theta}_k$'s. The exact definition of $d(\mathbf{\Theta})$ is beyond the scope of this paper but can be found in [13]. Let Ω denote the solution set

$$\Omega = \{\mathbf{\Theta}^* :\ d(\mathbf{\Theta}^*) = \max_{\mathbf{\Theta}} d(\mathbf{\Theta})\} \tag{17}$$

and let $\mathcal{P}_\Omega\mathbf{\Theta}$ denote the projection of $\mathbf{\Theta}$ onto the solution set Ω. From the derivations in [13] and duality theory [21] we know that (details omitted)

Theorem V.2. *Let $\mathbf{x}_k^{(i)}$ denote the eigenvector corresponding to the smallest eigenvalue of $\overline{\mathbf{R}}_k^{(i)} = \mathbf{R}_k + \mathbf{\Theta}_k^{(i)}$, and let $\mathbf{w}^*$ denote the distributed TLS solution (3). Let*

$$\mathbf{w}_k^{(i)} = -\frac{1}{\mathbf{e}_N^T \mathbf{x}_k^{(i)}} \left[\begin{array}{c|c} \mathbf{I}_P & \mathbf{0}_P \end{array} \right] \mathbf{x}_k^{(i)} \tag{18}$$

as computed by the D-TLS algorithm in Table I. Then $\lim_{i\to\infty} \mathbf{w}_k^{(i)} = \mathbf{w}^*$, $\forall\, k \in \mathcal{K}$ if and only if

$$\lim_{i\to\infty} \|\mathbf{\Theta}^{(i)} - \mathcal{P}_\Omega \mathbf{\Theta}^{(i)}\|_F = 0 \; . \tag{19}$$

Here, we implicitely assume that the $\mathbf{x}_k^{(i)}$'s are uniquely defined in the convergence point $\mathbf{\Theta}^{(\infty)}$, i.e., the smallest eigenvalue of $\overline{\mathbf{R}}_k^{(\infty)}$ does not degenerate, $\forall\, k \in \mathcal{K}$. If not, there exist multiple solutions at certain nodes, and then the nodes may have difficulties to reach a consensus (see also Subsection IV-B).

Define the generalized open ball around the solution set Ω with radius ϵ as

$$B(\Omega, \epsilon) = \{\mathbf{\Theta} \in \mathbb{R}^{KN\times N} : \; \|\mathbf{\Theta} - \mathcal{P}_\Omega \mathbf{\Theta}\|_F < \epsilon\} \; . \tag{20}$$

From [13], we also obtain the following theorem:

Theorem V.3. *For every dual optimal solution $\mathbf{\Theta}^* \in \Omega$, we have*

$$\begin{array}{c} \forall\, \epsilon_1 > 0,\; \exists\, \delta_1 > 0,\; \forall\, \mathbf{\Theta}^{(i)} \in \mathbb{R}^{JN\times N} \backslash B(\Omega, \epsilon_1) : \\ \mu_i \leq \delta_1 \Rightarrow \|\mathbf{\Theta}^{(i+1)} - \mathbf{\Theta}^*\|_F < \|\mathbf{\Theta}^{(i)} - \mathbf{\Theta}^*\|_F \end{array} \tag{21}$$

where $\mathbf{\Theta}^{(i+1)}$ is the outcome of a D-TLS iteration as described above.

This theorem can be straightforwardly proven by applying Theorem III.1 from [13] to the D-TLS subgradient update. Note that this theorem does not imply that any update (5) is in an ascent direction of $d(\mathbf{\Theta})$, i.e., in contrast to a gradient method, a subgradient method does not necessarily improve the objective function in each iteration. It is noted that Theorem V.3, together with (6)-(7) implies[2] (19), and therefore the D-TLS algorithm converges to the correct solution under the step size assumptions (6)-(7).

B. Eigenvector tracking

We first prove that the IPI-D-TLS algorithm can track the eigenvector of $\overline{\mathbf{R}}_k^{(i)} = \left(\overline{\mathbf{P}}_k^{(i)}\right)^{-1}$ corresponding to its smallest eigenvalue, i.e., the vectors $\mathbf{x}_k^{(i)}$ remain in an arbitrarily small neighborhood of the eigenvector $\mathbf{b}_{k,N}^{(i)}$ (note that this eigenvector changes with the iteration index i):

Theorem V.4. *Let $\mathbf{b}_{k,N}^{(i)}$ denote the eigenvector corresponding to the smallest eigenvalue of $\overline{\mathbf{R}}_k^{(i)}$ (i.e., the largest eigenvalue of $\overline{\mathbf{P}}_k^{(i)}$) and let Assumptions 2, 3 and 4 be satisfied, then*

$$\forall\, k \in \mathcal{K}, \forall\, \epsilon_2 > 0, \exists\, L_2 \in \mathbb{N} : \; i > L_2 \Rightarrow \|\mathbf{b}_{k,N}^{(i)} \pm \mathbf{x}_k^{(i)}\| < \epsilon_2 \tag{22}$$

where $\pm$ is used to resolve the sign ambiguity, i.e. either '$+$' or '$-$' is used, whichever gives the smallest norm.

Proof: In the sequel, let

$$\overline{\mathbf{P}}_k^{(i)} = \mathbf{Q}_k^{(i)} \mathbf{\Sigma}_k^{(i)} \mathbf{Q}_k^{(i)\,T} \tag{23}$$

denote the eigenvalue decomposition of $\overline{\mathbf{P}}_k^{(i)}$, where

$$\mathbf{\Sigma}_k^{(i)} = \text{diag}\left\{\sigma_{k,1}^{(i)}, \ldots, \sigma_{k,N}^{(i)}\right\} = \text{diag}\left\{\frac{1}{\lambda_{k,N}^{(i)}}, \ldots, \frac{1}{\lambda_{k,1}^{(i)}}\right\} \tag{24}$$

such that $\sigma_{k,1}^{(i)} \geq \sigma_{k,2}^{(i)} \geq \ldots \geq \sigma_{k,N}^{(i)}$.

We now analyze the inverse power iterations at an arbitrary node. For the sake of an easy exposition, we therefore omit the subscript k, and we keep in mind that the proof holds for all $k \in \mathcal{K}$. We work with the eigenvectors of the matrix $\overline{\mathbf{P}}^{(i)}$ which are the same as the eigenvectors of $\overline{\mathbf{R}}^{(i)}$, but with reversed indices, i.e., $\mathbf{b}_N^{(i)} = \mathbf{q}_1^{(i)}$ and so on. We will prove that

$$\forall\, \epsilon_2 > 0, \exists\, L_2 \in \mathbb{N} : \; i > L_2 \Rightarrow \|\mathbf{q}_1^{(i)} \pm \mathbf{x}^{(i)}\| < \epsilon_2 \; . \tag{25}$$

First, it is noted that Assumption 2 implies

$$\lim_{i\to\infty} \mu_i = 0 \; . \tag{26}$$

Define $\mathbf{E}^{(i)}$ such that $\overline{\mathbf{P}}^{(i+1)} = \overline{\mathbf{P}}^{(i)} + \mathbf{E}^{(i)}$. Because of Assumption 4 ($\lambda_N^{(i)} > \xi_1 > 0$, $\forall\, i \in \mathbb{N}$), expression (26) and the fact that $\|\mathbf{x}_q^{(i)}\| = 1$, $\forall\, q \in \mathcal{K}, \forall\, i \in \mathbb{N}$, we know that the entries of $\mathbf{E}^{(i)}$ become infinitesimaly small, i.e.,

$$\lim_{i\to\infty} \|\mathbf{E}^{(i)}\|_F = 0 \; . \tag{27}$$

Since the IPI method is not affected by an arbitrary scaling of the matrix $\overline{\mathbf{P}}^{(i)}$, or an arbitrary scaling of the iteration vector $\mathbf{x}^{(i)}$, we may replace the power iteration (13) with the following sequence of operations:

$$\mathbf{v}^{(i)} = \frac{1}{\mathbf{q}_1^{(i)\,T} \overline{\mathbf{x}}^{(i-1)}} \mathbf{Q}^{(i)\,T} \overline{\mathbf{x}}^{(i-1)} \tag{28}$$

$$\mathbf{u}^{(i)} = \frac{1}{\sigma_1^{(i)}} \mathbf{\Sigma}^{(i)} \mathbf{v}^{(i)} \tag{29}$$

$$\overline{\mathbf{x}}^{(i)} = \mathbf{Q}^{(i)} \mathbf{u}^{(i)} \; . \tag{30}$$

This modified IPI method produces a sequence of vectors $\{\overline{\mathbf{x}}^{(i)}\}_{i\in\mathbb{N}}$ that is exactly the same as the sequence $\{\mathbf{x}^{(i)}\}_{i\in\mathbb{N}}$ produced by the IPI-D-TLS algorithm, up to a scaling ambiguity, i.e.,

$$\mathbf{x}^{(i)} = \tau^{(i)} \overline{\mathbf{x}}^{(i)} \tag{31}$$

with $\tau^{(i)}$ a non-zero scalar. It is noted that the first entry of $\mathbf{v}^{(i)}$ and $\mathbf{u}^{(i)}$ are always equal to 1. Let

$$\mathbf{v}^{(i)} = \left[\begin{array}{c} 1 \\ \hline \tilde{\mathbf{v}}^{(i)} \end{array} \right] , \quad \mathbf{u}^{(i)} = \left[\begin{array}{c} 1 \\ \hline \tilde{\mathbf{u}}^{(i)} \end{array} \right] \; . \tag{32}$$

In the sequel, we will prove that $\lim_{i\to\infty} \|\tilde{\mathbf{u}}^{(i)}\| = 0$. Together

[2] The proof of this statement is omitted here since it is a special case of the proof of Theorem V.6, which is given later in this paper.

with (30) and (32), this proves (25), since $\|\tilde{\mathbf{u}}^{(i)}\| = 0$ implies that $\overline{\mathbf{x}}^{(i)} = \mathbf{q}_1^{(i)}$ and since $\|\mathbf{q}_1^{(i)}\| = 1$ we have $\mathbf{x}^{(i)} = \overline{\mathbf{x}}^{(i)}$.

Due to (16) and (29), we know that

$$\|\tilde{\mathbf{u}}^{(i)}\| \leq \beta \|\tilde{\mathbf{v}}^{(i)}\| \; . \tag{33}$$

We implicitly define the matrices $\mathbf{G}^{(i)}$ and $\widetilde{\mathbf{E}}^{(i)}$ that transform the matrix $\mathbf{Q}^{(i)}$ into $\mathbf{Q}^{(i+1)}$ in a particular way, i.e.

$$\{\mathbf{G}^{(i)}, \widetilde{\mathbf{E}}^{(i)}\} \triangleq \underset{\mathbf{G}^{(i)}, \widetilde{\mathbf{E}}^{(i)}}{\arg\min} \|\widetilde{\mathbf{E}}^{(i)}\|_F \tag{34}$$

$$\text{s.t.} \;\; \mathbf{Q}^{(i+1)} = \mathbf{Q}^{(i)} \begin{bmatrix} 1 & \mathbf{0}_{N-1}^T \\ \mathbf{0}_{N-1} & \mathbf{G}^{(i)} \end{bmatrix} + \widetilde{\mathbf{E}}^{(i)} \tag{35}$$

$$\mathbf{G}^{(i)} \mathbf{G}^{(i)\,T} = \mathbf{G}^{(i)\,T} \mathbf{G}^{(i)} = \mathbf{I}_{N-1} \; . \tag{36}$$

Basically, this constrained optimization problem defines an optimal rotation inside the $(N-1)$-dimensional eigenspace orthogonal to $\mathbf{q}_1^{(i)}$, such that the perturbations on each rotated eigenvector are small, i.e., of the same order of magnitude as $\|\mathbf{E}^{(i)}\|_F$, i.e.

$$\lim_{i \to \infty} \|\widetilde{\mathbf{E}}^{(i)}\|_F = \lim_{i \to \infty} \|\mathbf{E}^{(i)}\|_F = 0 \tag{37}$$

for which a proof can be found in Appendix B.

Define

$$\alpha^{(i)} \triangleq \frac{1}{\mathbf{q}_1^{(i)\,T} \overline{\mathbf{x}}^{(i-1)}} \; . \tag{38}$$

We know from (96) (see Appendix B) that

$$\lim_{i \to \infty} \|\mathbf{q}_1^{(i)} - \mathbf{q}_1^{(i-1)}\| = 0 \tag{39}$$

and therefore:

$$i \to \infty : \mathbf{q}_1^{(i)\,T} \overline{\mathbf{x}}^{(i-1)} = \mathbf{q}_1^{(i-1)\,T} \overline{\mathbf{x}}^{(i-1)} \; . \tag{40}$$

With this, and since $\mathbf{q}_1^{(i)\,T} \overline{\mathbf{x}}^{(i)} = 1$, $\forall i \in \mathbb{N}$, (this follows from (30) and (32)), we find

$$\lim_{i \to \infty} \alpha^{(i)} = 1 \; . \tag{41}$$

Note that $\mathbf{q}_1^{(i)\,T} \overline{\mathbf{x}}^{(i)} = 1$ does not imply that $\mathbf{q}_1^{(i)} = \overline{\mathbf{x}}^{(i)}$ since $\overline{\mathbf{x}}^{(i)}$ is not normalized to have a unity norm (otherwise, the theorem would have been proven at this point). Substituting (35) and (30) in (28) yields

$$\mathbf{v}^{(i+1)} = \alpha^{(i+1)} \left(\mathbf{y}^{(i)} + \widetilde{\mathbf{E}}^{(i)} \overline{\mathbf{x}}^{(i)} \right) \tag{42}$$

where

$$\mathbf{y}^{(i)} \triangleq \begin{bmatrix} 1 \\ \tilde{\mathbf{y}}^{(i)} \end{bmatrix} \triangleq \begin{bmatrix} 1 \\ \mathbf{G}^{(i)\,T} \tilde{\mathbf{u}}^{(i)} \end{bmatrix} \; . \tag{43}$$

Removing the first entry in (42) and applying the triangle inequality yields

$$\|\tilde{\mathbf{v}}^{(i+1)}\| \leq |\alpha^{(i+1)}| \left(\|\tilde{\mathbf{y}}^{(i)}\| + \|\widetilde{\mathbf{E}}^{(i)} \overline{\mathbf{x}}^{(i)}\| \right) \tag{44}$$

(we did not remove the first entry in the last term, which is not a problem since it is an upper bound). Because the Frobenius norm is submultiplicative, and since $\|\overline{\mathbf{x}}^{(i)}\| = \|\mathbf{u}^{(i)}\| = \sqrt{1 + \|\tilde{\mathbf{u}}^{(i)}\|^2} \leq 1 + \|\tilde{\mathbf{u}}^{(i)}\|$, we find that

$$\|\tilde{\mathbf{v}}^{(i+1)}\| \leq |\alpha^{(i+1)}| \left(\|\tilde{\mathbf{y}}^{(i)}\| + \|\widetilde{\mathbf{E}}^{(i)}\|_F \|\tilde{\mathbf{u}}^{(i)}\| + \|\widetilde{\mathbf{E}}^{(i)}\|_F \right) . \tag{45}$$

Since $\tilde{\mathbf{y}}^{(i)}$ is a rotated version of $\tilde{\mathbf{u}}^{(i)}$, we have

$$\|\tilde{\mathbf{y}}^{(i)}\| = \|\tilde{\mathbf{u}}^{(i)}\| \; . \tag{46}$$

Combining (33), (45) and (46) we find that

$$\|\tilde{\mathbf{u}}^{(i+1)}\| \leq \gamma^{(i)} \|\tilde{\mathbf{u}}^{(i)}\| + \delta_2^{(i)} \tag{47}$$

with

$$\gamma^{(i)} = \beta |\alpha^{(i+1)}| \left(1 + \|\widetilde{\mathbf{E}}^{(i)}\|_F \right) \tag{48}$$

$$\delta_2^{(i)} = \beta |\alpha^{(i+1)}| \|\widetilde{\mathbf{E}}^{(i)}\|_F \; . \tag{49}$$

We will prove that

$$\forall\, \epsilon_3 > 0, \exists\, L_4 \in \mathbb{N} : \; i > L_4 \Rightarrow \|\tilde{\mathbf{u}}^{(i)}\| < \epsilon_3 \tag{50}$$

which is equivalent to proving that $\lim_{i \to \infty} \|\tilde{\mathbf{u}}^{(i)}\| = 0$, which is what we aim for. Because of (37) and (41), we know that

$$\lim_{i \to \infty} \gamma^{(i)} = \beta < 1 \tag{51}$$

$$\lim_{i \to \infty} \delta_2^{(i)} = 0 \; . \tag{52}$$

Therefore, for every $\epsilon_3 > 0$ there should exist a corresponding $L_5 \in \mathbb{N}$ and an η that satisfies $0 < \eta < (1-\beta)\epsilon_3$ such that

$$i > L_5 \Rightarrow \; \delta_2^{(i)} < (1 - \gamma^{(i)})\epsilon_3 - \eta \; . \tag{53}$$

Notice that such an η only exists if L_5 is chosen large enough such that $i > L_5 \Rightarrow \; \gamma^{(i)} < 1$. From (47) and assuming that $i > L_5$ and $\|\tilde{\mathbf{u}}^{(i)}\| > \epsilon_3$, then

$$\|\tilde{\mathbf{u}}^{(i)}\| - \|\tilde{\mathbf{u}}^{(i+1)}\| \geq \left(1 - \gamma^{(i)}\right) \|\tilde{\mathbf{u}}^{(i)}\| - \delta_2^{(i)} \tag{54}$$

$$> \left(1 - \gamma^{(i)}\right) \epsilon_3 - \delta_2^{(i)} \tag{55}$$

$$> \eta \; . \tag{56}$$

Here, we used (47) in the first step, and (53) in the last step. Expression (56) shows that

$$i > L_5 \Rightarrow \|\tilde{\mathbf{u}}^{(i+1)}\| < \|\tilde{\mathbf{u}}^{(i)}\| - \eta \tag{57}$$

i.e., if $i > L_5$, $\|\tilde{\mathbf{u}}^{(i)}\|$ will shrink with a finite amount until $\|\tilde{\mathbf{u}}^{(i)}\| \leq \epsilon_3$. Therefore, there exists an $i = L_6 > L_5$ such that $\|\tilde{\mathbf{u}}^{(i)}\| \leq \epsilon_3$. We then only have to prove that once this holds, it holds for all future iterations, i.e., if $i > L_6$, then $\left(\|\tilde{\mathbf{u}}^{(i)}\| \leq \epsilon_3 \Rightarrow \|\tilde{\mathbf{u}}^{(i+1)}\| \leq \epsilon_3\right)$. Therefore, assume that $\|\tilde{\mathbf{u}}^{(i)}\| \leq \epsilon_3$ and $i > L_6$. From (47) and (53), we obtain

$$\|\tilde{\mathbf{u}}^{(i+1)}\| \leq \gamma^{(i)} \|\tilde{\mathbf{u}}^{(i)}\| + \delta_2^{(i)} \tag{58}$$

$$< \gamma^{(i)} \epsilon_3 + (1 - \gamma^{(i)})\epsilon_3 - \eta \tag{59}$$

$$< \epsilon_3 - \eta \tag{60}$$

and therefore $\|\tilde{\mathbf{u}}^{(i+1)}\|$ remains smaller than ϵ_3. Therefore, (50) holds by choosing $L_4 = L_6$. ■

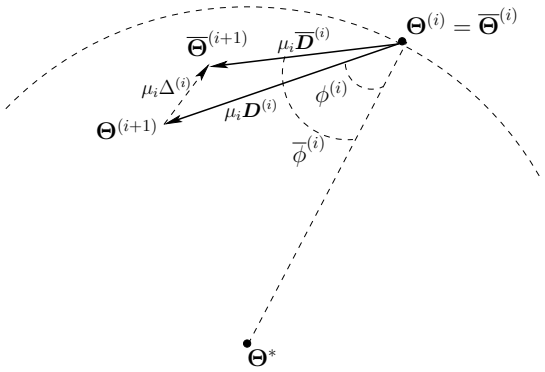

Fig. 1. Schematic illustration of the D-TLS update and the IPI-D-TLS update of $\boldsymbol{\Theta}^{(i)} = \overline{\boldsymbol{\Theta}}^{(i)}$.

C. Proof of convergence to solution set $\boldsymbol{\Omega}$

We define the variable $\overline{\boldsymbol{\Theta}}_k^{(i)} = \left(\overline{\mathbf{P}}_k^{(i)}\right)^{-1} - \mathbf{R}_k$ which corresponds to the $\boldsymbol{\Theta}_k^{(i)}$ as defined in Table I, but then with the $\mathbf{x}_k^{(i)}$'s and $\mathbf{x}_q^{(i)}$'s from the IPI-D-TLS algorithm instead of those from the D-TLS algorithm. Note that $\overline{\boldsymbol{\Theta}}_k^{(i)}$ is never explicitely computed in the IPI-algorithm. Similar to $\boldsymbol{\Theta}^{(i)}$, we define $\overline{\boldsymbol{\Theta}}^{(i)}$ as the stacked version of all the $\overline{\boldsymbol{\Theta}}_k^{(i)}$'s.

We will now prove two important theorems that will be needed in the sequel. The first theorem states that the IPI-D-TLS algorithm generates an infinite amount of points $\overline{\boldsymbol{\Theta}}^{(i)}$ that are within a chosen distance from the set Ω as defined in (17). The second theorem states that, if $\overline{\boldsymbol{\Theta}}^{(i)}$ gets within a chosen range from Ω and if i is sufficiently large, it will stay within this range in the next iteration.

Theorem V.5. *It holds that*

$$\forall\, \epsilon_4 > 0, \forall\, L_7 \in \mathbb{N}, \exists\, i_0 > L_7 : \overline{\boldsymbol{\Theta}}^{(i_0)} \in B(\Omega, \epsilon_4) \tag{61}$$

where $B(\Omega, \epsilon_4)$ denotes the generalized ball around Ω with radius ϵ_4, as defined in (20).

Proof: We will prove this theorem by reductio ad absurdum, i.e., we assume in the sequel that

$$\exists\, L_7 \in \mathbb{N}, \forall\, i > L_7 : \overline{\boldsymbol{\Theta}}^{(i)} \notin B(\Omega, \epsilon_4) \tag{62}$$

and we will show that this results in a contradiction.

Assume that we are in iteration i of the IPI-D-TLS algorithm, where we obtained $\overline{\boldsymbol{\Theta}}^{(i)} \notin B(\Omega, \epsilon_4)$. We define $\overline{\mathbf{D}}^{(i)}$ as the update direction of the IPI-D-TLS algorithm at iteration i, i.e.

$$\overline{\boldsymbol{\Theta}}^{(i+1)} = \overline{\boldsymbol{\Theta}}^{(i)} + \mu_i \overline{\mathbf{D}}^{(i)} \ . \tag{63}$$

Let $\mathbf{D}^{(i)}$ denote the update direction that the original D-TLS algorithm would apply if we set $\boldsymbol{\Theta}^{(i)} = \overline{\boldsymbol{\Theta}}^{(i)}$. The discrepancy between the update directions of the D-TLS algorithm and the IPI-D-TLS algorithm is then defined by

$$\Delta^{(i)} = \overline{\mathbf{D}}^{(i)} - \mathbf{D}^{(i)} \ . \tag{64}$$

This is schematically depicted in Fig. 1.

From Theorem V.4, we know that

$$\lim_{i \to \infty} \|\mathbf{x}_k^{(i)} - \mathbf{b}_{k,N}^{(i)}\| = 0 \tag{65}$$

where $\mathbf{x}_k^{(i)}$ is the vector as computed in step 3 of the IPI-D-TLS algorithm. It is noted that the corresponding $\mathbf{x}_k^{(i)}$'s, as computed in step 3 of the D-TLS algorithm when $\boldsymbol{\Theta}^{(i)} = \overline{\boldsymbol{\Theta}}^{(i)}$, are equal to the $\mathbf{b}_{k,N}^{(i)}$'s in Theorem V.4 since D-TLS computes exact eigenvectors. Since $\overline{\mathbf{D}}^{(i)}$ and $\mathbf{D}^{(i)}$ only depend on their respective $\mathbf{x}_k^{(i)}$'s (see update (5)), it straightforwardly follows from (65) that

$$\lim_{i \to \infty} \|\Delta^{(i)}\|_F = 0 \ . \tag{66}$$

It is tempting to conclude from (64) and (66) that the updates of the original D-TLS algorithm become equivalent to those of the IPI-D-TLS algorithm, and that convergence of the former therefore also implies convergence of the latter. However, this is a wrong conclusion at this point, since the norms of both $\overline{\mathbf{D}}^{(i)}$ and $\mathbf{D}^{(i)}$ become infinitely small when $\overline{\boldsymbol{\Theta}}^{(i)}$ gets closer to the solution set Ω. Therefore, even though $\|\Delta^{(i)}\|_F$ becomes infinitely small, the update directions $\overline{\mathbf{D}}^{(i)}$ and $\mathbf{D}^{(i)}$ may still be completely different.

Now choose a $\boldsymbol{\Theta}^* \in \Omega$, and define $\phi^{(i)}$ as illustrated in Fig. 1, i.e., $\phi^{(i)} = \phi\left(\boldsymbol{\Theta}^{(i)}\right)$ where $\phi(\boldsymbol{\Theta})$ is the operator that returns the angle between *vec* $(\boldsymbol{\Theta}^* - \boldsymbol{\Theta})$ and *vec* $(\mathbf{D}(\boldsymbol{\Theta}))$, where *vec*$(\mathbf{A})$ is the operator that creates a vector by stacking all the columns of the matrix $\mathbf{A}$, and with $\mathbf{D}(\boldsymbol{\Theta})$ denoting the subgradient $\mathbf{D}^{(i)}$ applied by the D-TLS algorithm when $\boldsymbol{\Theta}^{(i)} = \boldsymbol{\Theta}$. From Theorem V.3, we know that the subgradient applied by the D-TLS algorithm always points towards the inside of the sphere with center point $\boldsymbol{\Theta}^*$ and radius $\|\boldsymbol{\Theta}^* - \boldsymbol{\Theta}^{(i)}\|_F$, i.e.,

$$0 \leq |\phi^{(i)}| < \frac{\pi}{2} \ . \tag{67}$$

Because of convergence of the D-TLS algorithm[3] under Assumptions 1 and 2, we know that $\boldsymbol{\Theta}^{(i)}$ cannot diverge and therefore

$$\left\| \sum_{i=0}^{\infty} \mu_i \mathbf{D}^{(i)} \right\|_F < \infty \ . \tag{68}$$

In Appendix C we prove that

$$\sum_{i=0}^{\infty} \mu_i \|\Delta^{(i)}\|_F < \infty \ . \tag{69}$$

Using (68) and (69), we find that

$$\begin{aligned} \left\| \sum_{i=0}^{\infty} \mu_i \overline{\mathbf{D}}^{(i)} \right\|_F &= \left\| \sum_{i=0}^{\infty} \mu_i \left(\mathbf{D}^{(i)} + \Delta^{(i)} \right) \right\|_F \\ &\leq \left\| \sum_{i=0}^{\infty} \mu_i \mathbf{D}^{(i)} \right\|_F + \left\| \sum_{i=0}^{\infty} \mu_i \Delta^{(i)} \right\|_F \\ &\leq \left\| \sum_{i=0}^{\infty} \mu_i \mathbf{D}^{(i)} \right\|_F + \sum_{i=0}^{\infty} \mu_i \|\Delta^{(i)}\|_F < \infty \ . \end{aligned} \tag{70}$$

[3]This was explained in [13], based on convergence results in [17].

Expression (70) shows that also the variable $\overline{\mathbf{\Theta}}^{(i)}$ of the IPI-D-TLS algorithm cannot diverge, and therefore there exists a closed compact set $\mathcal{C}$ such that $\overline{\mathbf{\Theta}}^{(i)} \in \mathcal{C}$, $\forall\, i \in \mathbb{N}$.

Define the set $\widetilde{\mathcal{C}} = \mathcal{C} \backslash B(\Omega, \epsilon_4)$. Since $B(\Omega, \epsilon_4)$ is an open set, and since $\mathcal{C}$ is closed and compact, the set $\widetilde{\mathcal{C}}$ is also closed and compact. With this, define

$$\delta_3 = \max_{\mathbf{\Theta} \in \widetilde{\mathcal{C}}} \phi\left(\mathbf{\Theta}\right) \tag{71}$$

$$\delta_4 = \min_{\mathbf{\Theta} \in \widetilde{\mathcal{C}}} \|\mathbf{D}\left(\mathbf{\Theta}\right)\|_F \; . \tag{72}$$

Since $\widetilde{\mathcal{C}}$ is closed and compact, we know that δ_3 and δ_4 exist. Because of (67), we know that

$$\delta_3 < \frac{\pi}{2} \; . \tag{73}$$

From Theorem V.2 we know that $\mathbf{\Theta} \notin \Omega \Rightarrow \|\mathbf{D}\left(\mathbf{\Theta}\right)\|_F > 0$, and therefore

$$\delta_4 > 0 \; . \tag{74}$$

Since we have assumed that $\forall\, i > L_7 : \; \overline{\mathbf{\Theta}}^{(i)} \notin B(\Omega, \epsilon_4)$, we know that $\forall\, i > L_7 : \; \overline{\mathbf{\Theta}}^{(i)} \in \widetilde{\mathcal{C}}$. Therefore, and because of (66) and (73)-(74), we know that there also exists a $\overline{\delta}_3 < \frac{\pi}{2}$, $\overline{\delta}_4 > 0$ and a corresponding L_9 (chosen such that $L_9 > L_7$) such that

$$\forall\, i > L_9 : \; \|\overline{\mathbf{D}}^{(i)}\|_F > \overline{\delta}_4 > 0 \tag{75}$$

and

$$\forall\, i > L_9 : \; |\overline{\phi}^{(i)}| < \overline{\delta}_3 < \frac{\pi}{2} \tag{76}$$

where $\overline{\phi}^{(i)}$ denotes the angle between $vec\left(\overline{\mathbf{D}}^{(i)}\right)$ and $vec\left(\mathbf{\Theta}^* - \overline{\mathbf{\Theta}}^{(i)}\right)$ (see Fig. 1). Since $\overline{\mathbf{D}}^{(i)}$ is constructed from normalized vectors $\mathbf{x}_k^{(i)}$ (see (13)), we know that there exists a $\overline{\delta}_5 > 0$ such that

$$\forall\, i \in \mathbb{N} : \; \|\overline{\mathbf{D}}^{(i)}\|_F < \overline{\delta}_5 \; . \tag{77}$$

Define

$$\eta^{(i)} = \|\overline{\mathbf{\Theta}}^{(i)} - \mathbf{\Theta}^*\|_F^2 - \|\overline{\mathbf{\Theta}}^{(i+1)} - \mathbf{\Theta}^*\|_F^2 \tag{78}$$

which is equal to the decrease of the squared distance between $\overline{\mathbf{\Theta}}$ and $\mathbf{\Theta}^*$ achieved at iteration i. From a straightforward trigonometric derivation, based on Fig. 1, we find that

$$\eta^{(i)} = 2\mu_i \cos(\overline{\phi}^{(i)}) \|\overline{\mathbf{\Theta}}^{(i)} - \mathbf{\Theta}^*\|_F \|\overline{\mathbf{D}}^{(i)}\|_F - \mu_i^2 \|\overline{\mathbf{D}}^{(i)}\|_F^2 \; . \tag{79}$$

With the bounds $\overline{\delta}_3$, $\overline{\delta}_4$, and $\overline{\delta}_5$ (see (75)-(77)), we find from (79) that (assuming $i > L_9$):

$$\eta^{(i)} \geq \mu_i \left(2\epsilon_4 \cos(\overline{\delta}_3)\overline{\delta}_4 - \mu_i \overline{\delta}_5^2\right) \; . \tag{80}$$

From this, together with the fact that μ_i becomes infinitesimally small (see (26)), we can conclude that there exists an $L_{10} > L_9$ such that $i > L_{10} \Rightarrow \eta^{(i)} \geq 0$. Since we assumed in (62) that $\overline{\mathbf{\Theta}}$ never reaches $B(\Omega, \epsilon_4)$, this means that the infinite sum of all the $\eta^{(i)}$'s is finite, i.e.

$$\sum_{i=0}^{\infty} \eta^{(i)} < \infty \; . \tag{81}$$

However, by making the summation of both sides of (80), we conclude that

$$\sum_{i=L_9}^{\infty} \eta^{(i)} \geq 2\epsilon_4 \cos(\overline{\delta}_3)\overline{\delta}_4 \sum_{i=L_9}^{\infty} \mu_i - \overline{\delta}_5^2 \sum_{i=L_9}^{\infty} \mu_i^2 = \infty \tag{82}$$

where the latter follows from Assumption 1 and 2. This contradicts (81) and therefore (62) cannot be true, which proves (61). ∎

Theorem V.6. *It holds that*

$$\begin{gathered} \forall\, \epsilon_4 > 0, \exists\, L_7 \in \mathbb{N} : \\ \left(\overline{\mathbf{\Theta}}^{(i)} \in B(\Omega, \epsilon_4) \wedge i > L_7\right) \Rightarrow \overline{\mathbf{\Theta}}^{(i+1)} \in B(\Omega, \epsilon_4) \end{gathered} \tag{83}$$

where $\wedge$ denotes the logic 'and' operator.

Proof: The proof of (83) actually also follows from the arguments in the proof of Theorem V.5. We have basically shown that, for any radius ϵ_4 and sufficiently large i, $\overline{\mathbf{\Theta}}^{(i+1)}$ will be closer to $\mathbf{\Theta}^*$ compared to $\overline{\mathbf{\Theta}}^{(i)}$ as long as $\overline{\mathbf{\Theta}}^{(i)} \notin B(\Omega, \epsilon_4)$ (this follows from the fact that $\eta^{(i)}$, as defined in (78), is shown to be always positive under these conditions). This will also hold for a radius $\epsilon_5 < \epsilon_4$, i.e.

$$\begin{gathered} \forall\, L_{11} \in \mathbb{N},\; \exists\, L_{12} > L_{11} : \\ \left(i > L_{12} \wedge \overline{\mathbf{\Theta}}^{(i)} \notin B(\Omega, \epsilon_5)\right) \Rightarrow \eta^{(i)} \geq 0 \; . \end{gathered} \tag{84}$$

Now choose L_{11} such that

$$i > L_{11} \Rightarrow \mu_i \overline{\delta}_5 < \epsilon_4 - \epsilon_5 \; . \tag{85}$$

We know that such an L_{11} must exist because of (26). For this L_{11} choose a corresponding $L_{12} > L_{11}$ such that (84) is satisfied. Now, because of (63), (77) and (85), we know that

$$i > L_{11} \Rightarrow \|\overline{\mathbf{\Theta}}^{(i+1)} - \overline{\mathbf{\Theta}}^{(i)}\|_F < \epsilon_4 - \epsilon_5 \; . \tag{86}$$

Now choose L_7 in (83) equal to $L_7 = L_{12}$, and assume that $\overline{\mathbf{\Theta}}^{(i)} \in B(\Omega, \epsilon_4)$ and $i > L_7$, i.e., the righthand side of (83) holds. Then, either $\overline{\mathbf{\Theta}}^{(i)} \in B(\Omega, \epsilon_5)$ or $\overline{\mathbf{\Theta}}^{(i)} \in B(\Omega, \epsilon_4) \backslash B(\Omega, \epsilon_5)$. In the former case, $\overline{\mathbf{\Theta}}^{(i+1)} \in B(\Omega, \epsilon_4)$ due to (86), and in the latter case it also holds that $\overline{\mathbf{\Theta}}^{(i+1)} \in B(\Omega, \epsilon_4)$ due to (84). This proves (83). ∎

Remark III: The above proofs also show that Theorem V.3, together with (6)-(7), implies (19), as already mentioned in the preliminaries. Indeed, by replacing $\overline{\mathbf{\Theta}}^{(i)}$ with $\mathbf{\Theta}^{(i)}$ in (78) and below, we prove convergence of $\mathbf{\Theta}^{(i)}$ to $B(\Omega, \epsilon_4)$.

D. Proof of the main convergence theorem

From the obtained results, we can now prove Theorem IV.1:

Proof: Based on an induction argument on Theorem V.5 and Theorem V.6, we find that

$$\forall\, \epsilon_4 > 0,\; \exists\, L_7 \in \mathbb{N} : \; i > L_7 \Rightarrow \overline{\mathbf{\Theta}}^{(i)} \in B(\Omega, \epsilon_4) \; . \tag{87}$$

This is equivalent to

$$\lim_{i \to \infty} \|\overline{\mathbf{\Theta}}^{(i)} - \mathcal{P}_\Omega \overline{\mathbf{\Theta}}^{(i)}\|_F = 0 \; . \tag{88}$$

Let $\mathbf{b}_k^{(i)}$ denote the eigenvector corresponding to the small-

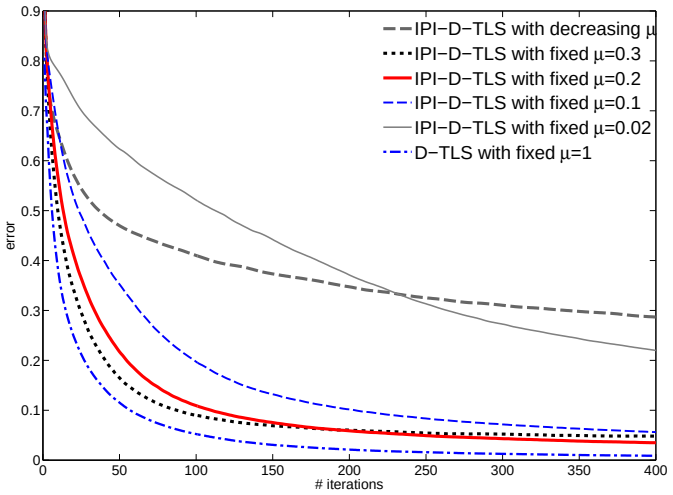

Fig. 2. Convergence properties of the D-TLS and IPI-D-TLS algorithm for different step sizes, averaged over 1000 Monte-Carlo runs.

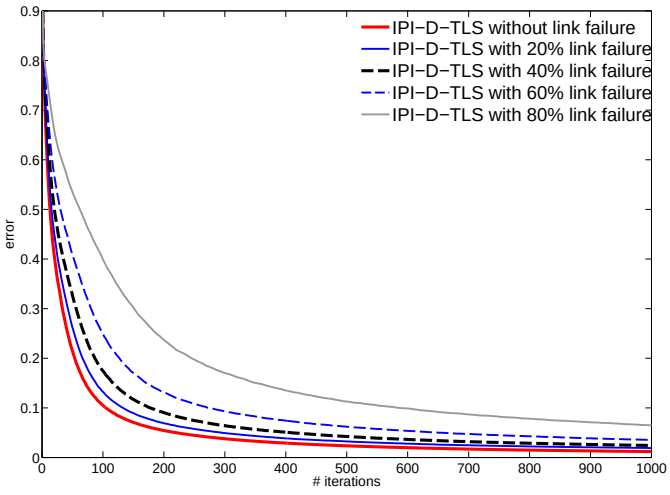

Fig. 3. Robustness of the IPI-D-TLS algorithm against random link failures, averaged over 1000 Monte-Carlo runs.

est eigenvalue of $\overline{\mathbf{R}}_k^{(i)} = \mathbf{R}_k + \overline{\mathbf{\Theta}}_k^{(i)}$, and let

$$\widetilde{\mathbf{w}}_k^{(i)} = -\frac{1}{\mathbf{e}_N^T \mathbf{b}_k^{(i)}} \left[\begin{array}{c|c} \mathbf{I}_P & \mathbf{0}_P \end{array} \right] \mathbf{b}_k^{(i)} . \tag{89}$$

Then we know from Theorem V.2 that, if (88) holds, the $\widetilde{\mathbf{w}}_k^{(i)}$'s, $\forall\, k \in \mathcal{K}$, converge to the distributed TLS solution (3). From Theorem V.4, we know that the $\mathbf{x}_k^{(i)}$'s in the IPI-D-TLS algorithm get infinitesimally close to the $\mathbf{b}_k^{(i)}$'s. Therefore, the $\mathbf{w}_k^{(i)}$'s of the IPI-D-TLS algorithm get infinitesimally close to the $\widetilde{\mathbf{w}}_k^{(i)}$'s defined in (89). Convergence of the latter to the distributed TLS solution (3) also proves convergence of IPI-D-TLS to this solution in each node.

■

VI. Simulation Results

In this section, we provide numerical simulation[4] results that demonstrate the convergence properties of the IPI-D-TLS algorithm. To illustrate the general behavior, we show results that are averaged over multiple Monte-Carlo (MC) runs. In each MC run, the following process is used to generate the network and the sensor observations:

1) Construct a P-dimensional vector $\mathbf{w}$ where the entries are drawn from a zero-mean normal distribution with unit variance.
2) Create a random connected network with K nodes, with 3 neighbors per node on average. To create this network, we start from a random tree to guarantee that the network is connected. Links are then randomly added until the average number of links per node equals 3.
3) For each node $k \in \mathcal{K}$:
   - Construct a $2P \times P$ input data matrix $\mathbf{U}_k$ where the entries are drawn from a zero-mean normal distribution with unit variance. The matrix $\mathbf{U}_k$ is then scaled by a random factor drawn from a uniform distribution on the unit interval (this models the different observation SNR at each node).
   - Compute $\mathbf{d}_k = \mathbf{U}_k \mathbf{w}$.
   - Add zero-mean white Gaussian noise to the entries in $\mathbf{U}_k$ and $\mathbf{d}_k$, with a standard deviation of 0.5.

In each experiment, we choose $K = 20$, $N = P + 1 = 10$. To assess the convergence and optimality of the algorithm, we use the error between the true solution and the local estimate, averaged over the K nodes in the network:

$$\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \|\mathbf{x}_k \pm \mathbf{x}^*\|_2 \tag{90}$$

where $\mathbf{x}^*$ is the normalized eigenvector corresponding to the smallest eigenvalue of $\mathbf{R} = \sum_{k \in \mathcal{K}} \mathbf{R}_k$, i.e., the eigenvector from which the network-wide TLS solution $\mathbf{w}^*$ can be extracted based on (3). The '$\pm$' is again used to resolve the sign ambiguity.

Fig 2 shows the convergence over 400 iterations of the IPI-D-TLS algorithm for different choices of the step size μ_i, averaged over 1000 MC runs. We use a fixed step size ranging from 0.02 to 0.3, which results in a converging algorithm even though Assumptions 1 and 2 are not satisfied. Larger step sizes often cause the algorithm not to converge. The figure also shows the results of a variable step size strategy where $\mu_i = 0.3\frac{1}{i^{0.7}}$, which satisfies Assumptions 1 and 2. It is observed that this results in an impractically slow convergence.

It is observed that the convergence speed strongly depends on the choice of the step size. The proper choice of the step size for the IPI-D-TLS algorithm is crucial to obtain a sufficiently fast convergence and sufficient accuracy (as it is also the case in the original D-TLS algorithm). Small step sizes yield slow convergence, but too large step sizes may yield unstable behavior or insufficient accuracy. It is still an open question how the optimal step size strategy can be determined on the fly.

The convergence of the D-TLS algorithm is also shown in Fig. 2 as a reference (with fixed step size $\mu = 1$, which was empirically found to give the best convergence properties for

[4]Matlab code to reproduce these simulations can be found in *http://homes.esat.kuleuven.be/~abertran*.

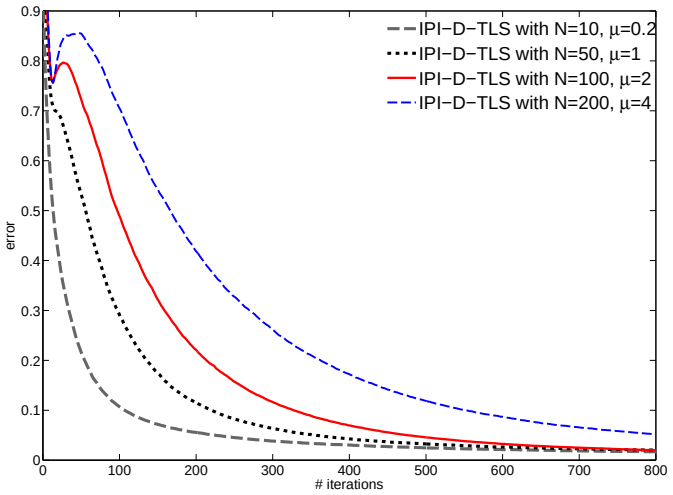

Fig. 4. Convergence properties of the IPI-D-TLS algorithm for different values of N, averaged over 500 Monte-Carlo runs.

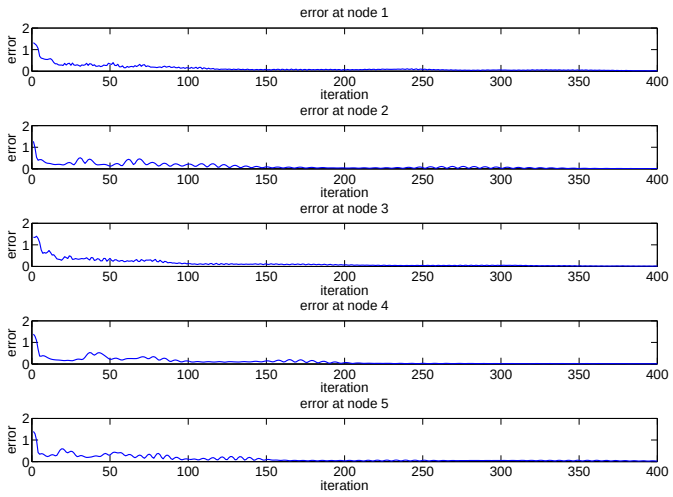

Fig. 5. Convergence properties of the IPI-D-TLS algorithm for $N = 200$ and $\mu = 2$, for nodes $k \in \{1, \ldots, 5\}$.

this scenario [13]). It is observed that the D-TLS algorithm converges faster than the IPI-D-TLS algorithm. This demonstrates that the original D-TLS algorithm has better tracking capabilities than the IPI-D-TLS algorithm. However, the true strength of the IPI-D-TLS algorithm is its low computational complexity, especially so for the estimation of large regressors, where it is too complex to calculate an exact EVD.

Fig. 3 shows the robustness of the IPI-D-TLS algorithm against random link failures. In each iteration of the algorithm, each link has a probability of failing equal to $p\%$, where p is chosen as $p \in \{0, 20, 40, 60, 80\}$. Basically, this means that the set of neighbors of node k can now change in each iteration, i.e., $\mathcal{N}_k$ is replaced by $\mathcal{N}_k^{(i)}$ in the IPI-D-TLS algorithm, where $\mathcal{N}_k^{(i)}$ contains all the neighbors of node k that are able to transmit their data to node k in iteration i. The results in Fig. 3 are averaged over 1000 MC runs. It is observed that the IPI-D-TLS algorithm is quite robust against random link failures, which is a property inherited from the D-TLS algorithm [13].

Fig. 4 shows the first 800 iterations of the IPI-D-TLS algorithm, averaged over 500 MC runs, for different $N = P + 1$, i.e., for increasing dimension of the D-TLS problem. In [13] it was shown that the maximum step size is more or less linearly dependent on the smallest eigenvalue of the matrix $\mathbf{R}$. Therefore, the step size is chosen as $\mu = 2N/100$, since the eigenvalues of the matrix $\mathbf{R}$ increase with N (remember that each node collects $2N$ observations). We do not show the results for the D-TLS algorithm since the processing time for $N \geq 100$ is too long.

Fig. 5 shows the error at 5 nodes in the network for a single run in a scenario where $N = 200$. The dynamics of the estimate $\mathbf{w}^{(i)}$ in this experiment (or other random scenarios) are visualized in a Matlab animation, which is provided as an attachement to this paper[5].

Finally, Fig. 6 shows the entries of the final estimate at node 1 in the same scenario as shown in Fig. 5, after 400 iterations. It is observed that the final estimate is quasi identical

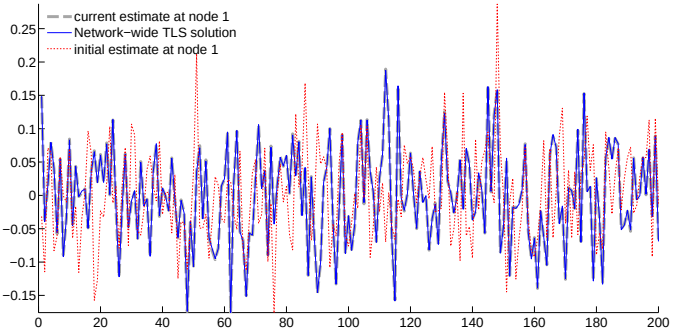

Fig. 6. The 200 entries of the final estimate $\mathbf{x}_1^{(400)}$ at node 1 after 400 iterations of the IPI-D-TLS algorithm for a scenario with $N = 200$ and $\mu = 2$.

to the network-wide TLS solution, and very different from the initial local TLS solution computed at node 1 without using the information from its neighbors.

VII. Conclusions

In this paper, we have revisited the D-TLS algorithm for linear regression problems in ad hoc wireless sensor networks. We have modified the algorithm to reduce its computational complexity, by replacing its EVD by a single IPI. We have shown that, even though the nodes do not share their true eigenvectors, the IPI-D-TLS algorithm still converges to the network-wide TLS solution under certain assumptions, which are often satisfied in practice. We have provided simulation results to demonstrate the convergence of the algorithm, even when some of these assumptions are not satisfied (e.g. when a fixed step size is used instead of a decreasing step size). As an attachement to this paper, we have provided Matlab scripts to visualize the dynamics of the algorithm. In future work, we will investigate whether the performance of IPI-D-TLS can be improved by applying diffusion techniques as in [7]–[9].

[5]This can also be found on *http://homes.esat.kuleuven.be/~abertran/software.html*.

APPENDIX

A. Proof of Lemma V.1

Note that, from the definition of $\overline{\mathbf{\Theta}}_k^{(i)}$, as defined in Subsection V-C, we have that

$$\forall\, i \in \mathbb{N} : \sum_{k\in\mathcal{K}} \overline{\mathbf{\Theta}}_k^{(i)} = \mathbf{O}_{N\times N}\ . \tag{91}$$

This means that, if the largest eigenvalue $\lambda_{k,1}^{(i)}$ of $\overline{\mathbf{R}}_k^{(i)}$ goes to infinity, then there should exist a node $q \neq k$ for which the smallest eigenvalue $\lambda_{q,N}^{(i)}$ goes to minus infinity. Since we have assumed that $\lambda_{q,N}^{(i)} > \xi_1 > 0$ (Assumption 4), this cannot hold, and therefore (15) must be true.

Since $\lambda_{k,N-1}^{(i)} - \lambda_{k,N}^{(i)} > \kappa_1$ if $i > L_1$ (Assumption 3) we find that

$$i > L_1 \Rightarrow 1 - \frac{\lambda_{k,N}^{(i)}}{\lambda_{k,N-1}^{(i)}} > \frac{\kappa_1}{\lambda_{k,N-1}^{(i)}}\ . \tag{92}$$

Since $\kappa_1 > 0$ and $0 < \lambda_{k,N-1}^{(i)} < \xi_2$, $\forall\, i \in \mathbb{N}$, we know that $\frac{\kappa_1}{\lambda_{k,N-1}^{(i)}} > \frac{\kappa_1}{\xi_2}$ and therefore

$$i > L_1 \Rightarrow \frac{\lambda_{k,N}^{(i)}}{\lambda_{k,N-1}^{(i)}} < 1 - \frac{\kappa_1}{\xi_2}\ . \tag{93}$$

By choosing $\beta = 1 - \frac{\kappa_1}{\xi_2}$, and since by definition $\kappa_1 < \xi_2$, also (16) is proven. ∎

B. Proof of (37):

Because of (27) we know there exists an L_3 such that

$$i > L_3 \Rightarrow \|\mathbf{E}^{(i)}\|_F \leq \frac{d^{(i)}}{4} \tag{94}$$

where $d^{(i)} = \sigma_1^{(i)} - \sigma_2^{(i)}$. Note that for the L_1 from Assumption 3, there also exists a $\kappa_2 > 0$ such that $i > L_1 \Rightarrow d^{(i)} > \kappa_2 > 0$, which follows from the upper and lower bounds ξ_1 and ξ_2 (see Lemma V.1) on the eigenvalues of $\overline{\mathbf{R}}_k^{(i)}$. Therefore, we choose $L_3 > L_1$. From [16, p. 399, Theorem 8.1.12], we know that, if the righthand side of (94) holds, then (details are omitted)

$$\sqrt{1 - \left(\mathbf{q}_1^{(i)\,T}\mathbf{q}_1^{(i+1)}\right)^2} \leq \frac{4}{d^{(i)}}\|\mathbf{E}^{(i)}\|_F\ . \tag{95}$$

With (27), and the fact that $d^{(i)} > \kappa_2 > 0$ for $i > L_3$, we therefore find that

$$\lim_{i\to\infty} \|\mathbf{q}_1^{(i)} \pm \mathbf{q}_1^{(i+1)}\| = 0\ . \tag{96}$$

Define $\mathbf{Q}_{-1}^{(i)}$ such that $\mathbf{Q}^{(i)} = \left[\mathbf{q}_1^{(i)}\ \mathbf{Q}_{-1}^{(i)}\right]$. Because of (96), the orthogonal complement of $\mathbf{q}_1^{(i+1)}$, i.e., the subspace spanned by the columns of $\mathbf{Q}_{-1}^{(i+1)}$, will get infinitesimally close to the subspace spanned by the columns of $\mathbf{Q}_{-1}^{(i)}$. In other words, there exists a rotation matrix $\mathbf{G}^{(i)}$ such that

$$\lim_{i\to\infty} \|\mathbf{Q}_{-1}^{(i)}\mathbf{G}^{(i)} - \mathbf{Q}_{-1}^{(i+1)}\|_F = 0\ . \tag{97}$$

Finally, (37) is proven by comparing (96) and (97) with (35). ∎

C. Proof of (69):

First, observe in the rank-1 update function (10), that $\|\mathbf{E}^{(i)}\|_F$ is $O(\mu_i)$, which means that

$$\exists\, C > 0, \exists\, M \in \mathbb{N} :\ i > M \Rightarrow \|\mathbf{E}^{(i)}\|_F < C\mu_i\ . \tag{98}$$

Using this in (95), and since $d^{(i)}$ is lower bounded with a positive value κ_2, we know that $\sin(\theta^{(i)})$ is $O(\mu_i)$, where $\theta^{(i)}$ denotes the angle between the new eigenvector $\mathbf{q}_1^{(i+1)}$ and the previous eigenvector $\mathbf{q}_1^{(i)}$ (to resolve the sign ambiguity, we choose $0 \leq \theta^{(i)} \leq \frac{\pi}{2}$). Based on a straightforward trigoniometric derivation (note that $\mathbf{q}_1^{(i)}$ and $\mathbf{q}_1^{(i+1)}$ are normalized to unity norm), we find that $\|\mathbf{q}_1^{(i)} - \mathbf{q}_1^{(i+1)}\| = 2\sin(\theta^{(i)}/2)$. Using the fact that $\sin(\theta^{(i)}/2) \leq \sin(\theta^{(i)})$ if $0 \leq \theta^{(i)} \leq \frac{\pi}{2}$, the above shows that $\|\mathbf{q}_1^{(i)} - \mathbf{q}_1^{(i+1)}\|$ is also $O(\mu_i)$. With this, and based on a similar reasoning as in Appendix B, we find that $\|\widetilde{\mathbf{E}}^{(i)}\|_F$ (with $\widetilde{\mathbf{E}}^{(i)}$ defined in (34)-(36)) is also $O(\mu_i)$.

Let us now focus on (47). Based on (51), and for the sake of an easy exposition, we will replace $\gamma^{(i)}$ with the fixed $\beta < 1$. Although this is not entirely correct, we know that if i is sufficiently large, this approximation becomes arbitrarily accurate. It is noted that the proof can be continued without making this replacement, but it will be more elaborate. From (41) and (49), and because $\|\widetilde{\mathbf{E}}^{(i)}\|_F$ is $O(\mu_i)$, $\delta_2^{(i)}$ is also $O(\mu_i)$. Therefore, (47) becomes

$$\exists C_1 > 0, \exists L_8 : i \geq L_8 \Rightarrow \|\tilde{\mathbf{u}}^{(i+1)}\| \leq \beta\|\tilde{\mathbf{u}}^{(i)}\| + C_1\mu_i\ . \tag{99}$$

Note that, since $\mu_i > 0$, $\forall i \in \mathbb{N}$, we can always choose $L_8 = 0$ by choosing C_1 large enough, so we will assume that $L_8 = 0$ in the sequel for the sake of an easy exposition. By expanding the iteration (99), we find (for $i > 0$)

$$\|\tilde{\mathbf{u}}^{(i)}\| \leq \beta^i\|\tilde{\mathbf{u}}^{(0)}\| + C_1\sum_{j=0}^{i}\beta^{i-j}\mu_j\ . \tag{100}$$

By making a summation over both sides of (100), we find that

$$\sum_{i=0}^{\infty}\mu_i\|\tilde{\mathbf{u}}^{(i)}\| \leq \|\tilde{\mathbf{u}}^{(0)}\|\sum_{i=0}^{\infty}\mu_i\beta^i + C_1\sum_{i=0}^{\infty}\mu_i\sum_{j=0}^{i}\beta^{i-j}\mu_j\ . \tag{101}$$

The sequence $(\beta^i)_{i\in\mathbb{N}}$ is $O(\mu_i)$ since $\sum_{i=0}^{\infty}\beta^i < \infty$ and $\sum_{i=0}^{\infty}\mu_i = \infty$. Therefore, and since the sequence $(\mu_i)_{i\in\mathbb{N}}$ is square-summable (Assumption 2), we can conclude that the first summation on the righthand side of (101) is bounded, i.e. there exists a C_2 such that

$$\sum_{i=0}^{\infty}\mu_i\|\tilde{\mathbf{u}}^{(i)}\| \leq C_2 + C_1\sum_{i=0}^{\infty}\mu_i\sum_{j=0}^{i}\beta^{i-j}\mu_j\ . \tag{102}$$

Note that the second term can be rewritten as

$$\begin{aligned}\sum_{i=0}^{\infty}\mu_i\sum_{j=0}^{i}\beta^{i-j}\mu_j &= \sum_{i=0}^{\infty}\mu_i^2 + \beta\sum_{i=0}^{\infty}\mu_i\mu_{i-1} + \beta^2\sum_{i=0}^{\infty}\mu_i\mu_{i-2} + \dots \\ &= \sum_{j=0}^{\infty}\beta^j\left(\sum_{i=0}^{\infty}\mu_i\mu_{i-j}\right)\end{aligned} \tag{103}$$

where we assume that $\mu_i = 0$ if $i < 0$. Define the j-shifted sequence $(\mu_i^j)_{i\in\mathbb{N}}$ such that $\mu_i^j = \mu_{i-j}$. Note that all sequences

$(\mu_i^j)_{i\in\mathbb{N}}$, $\forall\, j \in \mathbb{N}$, belong to l^2, i.e. the space of square-summable sequences. Since l^2 is an inner-product space, we can apply the Cauchy-Schwartz inequality to find that

$$\forall\, j \in \mathbb{N}:\ \left(\sum_{i=0}^{\infty} \mu_i \mu_{i-j}\right)^2 \leq \left(\sum_{i=0}^{\infty} \mu_i^2\right)\left(\sum_{i=0}^{\infty} \mu_{i-j}^2\right) \quad (104)$$

$$= \left(\sum_{i=0}^{\infty} \mu_i^2\right)^2 . \quad (105)$$

Substituting this in (103) yields

$$\sum_{i=0}^{\infty} \mu_i \sum_{j=0}^{i} \beta^{i-j}\mu_j \leq \left(\sum_{j=0}^{\infty} \beta^j\right)\left(\sum_{i=0}^{\infty} \mu_i^2\right) = \frac{1}{1-\beta}\sum_{i=0}^{\infty} \mu_i^2 . \quad (106)$$

Substituting this result in (102), and again relying on the fact that $(\mu_i)_{i\in\mathbb{N}}$ is square-summable (Assumption 2), we find that

$$\sum_{i=0}^{\infty} \mu_i \|\tilde{\mathbf{u}}^{(i)}\| < \infty . \quad (107)$$

Note that we have removed the subscript k in the proof of Theorem V.4 since the proof holds for an arbitrary node $k \in \mathcal{K}$, and therefore we actually have that $\sum_{i=0}^{\infty} \mu_i \|\tilde{\mathbf{u}}_k^{(i)}\| < \infty$, $\forall k \in \mathcal{K}$. Since $\|\Delta^{(i)}\|_F$ is $O(\sum_{k\in\mathcal{K}} \|\tilde{\mathbf{u}}_k^{(i)}\|)$ (this follows from expressions (30)-(32) and the (implicit) definition of $\overline{\mathbf{D}}^{(i)}$ in function of the $\mathbf{x}_k^{(i)}$'s), we eventually obtain

$$\exists\, C_3:\ \sum_{i=0}^{\infty} \mu_i \|\Delta^{(i)}\|_F \leq C_3 \sum_{k\in\mathcal{K}} \sum_{i=0}^{\infty} \mu_i \|\tilde{\mathbf{u}}_k^{(i)}\| < \infty \quad (108)$$

which proves (69). ■

REFERENCES

[1] A. Bertrand and M. Moonen, "Power iteration-based distributed total least squares estimation in ad hoc sensor networks," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Kyoto, Japan, March 2012.

[2] I. Markovsky and S. Van Huffel, "Overview of total least-squares methods," *Signal Processing*, vol. 87, no. 10, pp. 2283 – 2302, 2007.

[3] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks – part I: sequential node updating," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5277–5291, 2010.

[4] ——, "Distributed adaptive estimation of node-specific signals in wireless sensor networks with a tree topology," *IEEE Transactions on Signal Processing*, vol. 59, no. 5, pp. 2196–2210, 2011.

[5] ——, "Robust distributed noise reduction in hearing aids with external acoustic sensor nodes," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, Article ID 530435, 14 pages, 2009. doi:10.1155/2009/530435.

[6] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Processing*, vol. 55, no. 8, pp. 4064–4077, Aug. 2007.

[7] ——, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Processing*, vol. 56, no. 7, pp. 3122–3136, July 2008.

[8] F. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Processing*, vol. 56, no. 5, pp. 1865–1877, May 2008.

[9] A. Bertrand, M. Moonen, and A. H. Sayed, "Diffusion bias-compensated RLS estimation over adaptive networks," *IEEE Transactions on Signal Processing*, 2011.

[10] G. Mateos, I. D. Schizas, and G. Giannakis, "Closed-form MSE performance of the distributed LMS algorithm," in *Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop, 2009. DSP/SPE 2009. IEEE 13th*, 4-7 2009, pp. 66 –71.

[11] G. Mateos, I. D. Schizas, and G. B. Giannakis, "Performance analysis of the consensus-based distributed LMS algorithm," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, Article ID 981030, 19 pages, 2009. doi:10.1155/2009/981030.

[12] ——, "Distributed recursive least-squares for consensus-based in-network adaptive estimation," *IEEE Trans. Signal Processing*, vol. 57, no. 11, pp. 4583–4588, 2009.

[13] A. Bertrand and M. Moonen, "Consensus-based distributed total least squares estimation in ad hoc wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 59, no. 5, pp. 2320–2330, 2011.

[14] E. Dall'Anese and G. Giannakis, "Distributed cognitive spectrum sensing via group sparse total least-squares," in *IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, dec. 2011, pp. 341 –344.

[15] C. E. Davila, "An efficient recursive total least squares algorithm for FIR adaptive filtering," *IEEE Transactions on Signal Processing*, vol. 42, no. 2, pp. 267–280, 1994.

[16] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed. Baltimore: The Johns Hopkins University Press, 1996.

[17] D. Bertsekas, A. Nedic, and A. Ozdaglar, *Convex Analysis and Optimization*. Athena Scientific, 2003.

[18] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks – part II: simultaneous & asynchronous node updating," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5292–5306, 2010.

[19] P. Di Lorenzo and S. Barbarossa, "Bio-inspired swarming models for decentralized radio access incorporating random links and quantized communications," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, May 2011, pp. 5780 – 5783.

[20] M. Nevelson and R. Hasminskii, *Stochastic Approximation and Recursive Estimation*. Providence, Rhode Island: American Mathematical Society, 1973.

[21] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.



Alexander Bertrand (M'08) received the M.Sc. degree (2007) and the Ph.D. degree (2011) in Electrical Engineering, both from KU Leuven, University of Leuven, Belgium. He is currently a Postdoctoral Fellow of the Research Foundation - Flanders (FWO), affiliated with the Electrical Engineering Department (ESAT) of KU Leuven and the IBBT Future Health Department. In 2010, he was a visiting researcher at the Adaptive Systems Laboratory, University of California, Los Angeles (UCLA). His research interests are in multi-channel signal processing, ad hoc sensor arrays, wireless sensor networks, distributed signal enhancement, speech enhancement, and distributed estimation.

Dr. Bertrand received a Postdoctoral Fellowship from the Research Foundation - Flanders (FWO) (2011-2014), a Ph.D. scholarship of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen) (2008-2011), and an FWO grant for a Visiting Research Collaboration at UCLA (2010). He has served as a Technical Programme Committee (TPC) Member for the European Signal Processing Conference (EUSIPCO) 2012.



Marc Moonen (M'94, SM'06, F'07) received the electrical engineering degree and the PhD degree in applied sciences from KU Leuven, Belgium, in 1986 and 1990 respectively.

Since 2004 he is a Full Professor at the Electrical Engineering Department of KU Leuven, where he heads a research team working in the area of numerical algorithms and signal processing for digital communications, wireless communications, DSL and audio signal processing.

He received the 1994 KU Leuven Research Council Award, the 1997 Alcatel Bell (Belgium) Award (with P. Vandaele), the 2004 Alcatel Bell (Belgium) Award (with R. Cendrillon), and was a 1997 "Laureate of the Belgium Royal Academy of Science". He received a journal best paper award from the *IEEE Transactions on Signal Processing* (with G. Leus) and from *Elsevier Signal Processing* (with S. Doclo).

He was chairman of the IEEE Benelux Signal Processing Chapter (1998-2002), and a member of the IEEE Signal Processing Society Technical Committee on Signal Processing for Communications, and is currently President of EURASIP (European Association for Signal Processing).

He served as Editor-in Chief for the *EURASIP Journal on Applied Signal Processing* (2003-2005), and has been a member of the editorial board of *IEEE Transactions on Circuits and Systems II*, *IEEE Signal Processing Magazine*, *Integration-the VLSI Journal*, *EURASIP Journal on Wireless Communications and Networking*, and *Signal Processing*. He is currently a member of the editorial board of *EURASIP Journal on Applied Signal Processing*, and Area Editor for Feature Articles in *IEEE Signal Processing Magazine*.