DISTRIBUTED ESTIMATION OF CROSS-CORRELATION FUNCTIONS IN AD-HOC MICROPHONE ARRAYS

Toon van Waterschoot

KU Leuven, Department of Electrical Engineering (ESAT-STADIUS/ETC)
Kasteelpark Arenberg 10, 3001 Leuven, Belgium
toon.vanwaterschoot@esat.kuleuven.be

ABSTRACT

In this paper, we address the problem of estimating the cross-correlation function between two microphone signals recorded in different nodes of an ad-hoc microphone array or wireless acoustic sensor network, where the transmission of the entire microphone signal from one node to another is undesirable due to power and/or bandwidth constraints. We show that instead of directly computing the cross-correlation function, it can be estimated as the solution to a deconvolution problem. This deconvolution problem can be separated into two subproblems, each of which depends on one microphone signal and an auxiliary signal derived from the other microphone signal. Three different strategies for solving this deconvolution problem are proposed, in which the two subproblems are solved jointly (symmetric deconvolution), separately (asymmetric deconvolution) or in a consensus framework (consensus deconvolution). Simulation results illustrate the performance difference in terms of estimation accuracy, noise robustness, and transmission requirements.

***Index Terms*—** distributed signal processing, distributed optimization, cross-correlation function, ad-hoc microphone array, wireless acoustic sensor network

1. INTRODUCTION

Ad-hoc microphone arrays and wireless acoustic sensor networks (WASNs) have recently gained an increasing interest in the signal processing community [1]. Such arrays and networks provide a flexible way of connecting a multitude of microphones, often residing in different physical devices, with the aim of covering larger spatial areas and increasing spatial diversity. Multi-microphone signal processing in ad-hoc arrays and WASNs is more challenging than in traditional microphone arrays, due to distributed nature of the signal acquisition and processing. In ad-hoc arrays and WASNs, microphones are usually connected through wireless radio links and hence the transmission of signals or parameters from one microphone to another requires power and bandwidth, which is often costly.

The design of multi-microphone signal processing algorithms for ad-hoc arrays and WASNs is therefore rooted in distributed signal processing and distributed optimization theory. A distributed design approach is often based on a (approximate or exact) reformulation of the original, so-called global problem into a number of interrelated subproblems, each of which can be solved in one of the array or network nodes using local microphone signals and a limited number of data shared by other nodes. Here, a node consists of one or more microphones assembled in the same device, a local processing unit, and means for wireless radio communication. Two major challenges in this design approach are (1) to define the interrelation between subproblems in such a way that the amount of data (i.e., signal samples or parameter values) to be shared among different nodes is minimized without compromising the solution to the global problem, and (2) to derive an efficient algorithm to solve the reformulated problem in a distributed fashion. Two types of distributed estimation problems are often encountered in ad-hoc arrays and WASNs: (1) *node-specific* estimation problems in which each node aims to estimate a different signal or parameter vector [2], and (2) *consensus* estimation problems in which all nodes aim to estimate the same signal or parameter vector [3].

In this paper, we consider the problem of estimating the cross-correlation function between two signals acquired by microphones in different array or network nodes. Microphone signal cross-correlation functions are needed in a wide variety of multi-microphone signal processing algorithms such as time delay estimation, source localization, microphone self-localization, beamforming, and (relative) acoustic transfer function estimation, hence the estimation of cross-correlation functions is of key importance. However, the direct computation of cross-correlation functions in ad-hoc arrays and WASNs would require the full transmission of raw microphone signals within the network, which is highly undesirable as mentioned before. Our objective is therefore to

This research work was carried out at the ESAT Laboratory of KU Leuven, and was supported by KU Leuven Research Council CoE PFV/10/002 (OPTEC), KU Leuven Impulse Fund IMP/14/037, and the FP7-PEOPLE Marie Curie Initial Training Network "Dereverberation and Reverberation of Audio, Music, and Speech (DREAMS)", funded by the European Commission under Grant Agreement no. 316969. The scientific responsibility is assumed by its authors.

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

reformulate the cross-correlation function estimation problem in such a way that it can be efficiently solved in a distributed manner. To the best of our knowledge, this particular problem has not yet been considered in the literature. Two different but somewhat related problems have recently been addressed in [4, 5]. In [4], the distributed estimation of the inverse short-time Fourier transform (STFT)-domain correlation matrix of a multi-microphone signal vector is solved by means of an algorithm requiring two distributed averaging operations (performed by means of gossip algorithms) in each STFT bin. In [5], the generalized eigenvectors corresponding to the Q largest (or smallest) generalized eigenvalues of a pair of sensor signal covariance matrices are estimated in a distributed fashion, using an algorithm in which each node updates one part of the Q-column matrix of eigenvectors and broadcasts a compressed version of its sensors signals to all other nodes.

The starting point for the proposed distributed approach to estimate the cross-correlation function between two signals, is the observation that the convolution of the cross-correlation function with one of both signals is equal to the convolution of the autocorrelation function of that signal with the other signal. This equivalence allows to reformulate the cross-correlation estimation problem as a deconvolution problem. Moreover, this deconvolution problem can be separated into two subproblems, each of which depends on one of both microphone signals and on a limited set of samples from an auxiliary signal derived from the other microphone signal. Given that each of the subproblems can be solved relatively easily, we propose three approaches for solving the global deconvolution problem. In the *symmetric deconvolution* approach, both subproblems are jointly solved, requiring the additional transmission of a number of microphone signal samples between the two nodes. In the *asymmetric deconvolution* approach, we simply solve each subproblem in one node, resulting in different estimates of the same cross-correlation function in different nodes. Finally, in the *consensus deconvolution* approach, we again solve each subproblem in one node, but by including consensus constraints we force the different subproblem solutions to be equal. This consensus problem is then solved by means of the alternating direction method of multipliers (ADMM) [6].

The paper is organized as follows. In Section 2, we show how the cross-correlation estimation problem can be reformulated as a deconvolution problem. Section 3 introduces three different approaches and algorithms to solve the deconvolution problem in a distributed manner. In Section 4, simulation results are presented aiming at a performance comparison of the three proposed approaches with the direct approach to cross-correlation estimation in terms of estimation accuracy, noise robustness, and transmission requirements.

2. DECONVOLUTION APPROACH TO CROSS-CORRELATION ESTIMATION

We consider a pair of microphones indexed by (i, j) in an ad-hoc microphone array or WASN, where microphone i resides in a different node[1] than microphone j. These microphones acquire the signals $y_i(n) = x_i(n) + v_i(n)$ and $y_j(n) = x_j(n) + v_j(n)$, respectively, for $n = 1, \ldots, N$. Here, $x_i(n), x_j(n)$ represent the correlated components (due to near-field or far-field sound sources) and $v_i(n), v_j(n)$ represent the uncorrelated components (e.g., sensor noise or diffuse acoustic noise) in the microphone signals. It is assumed that all signals are stationary and ergodic in the observation interval $[1, N]$. The aim is to estimate the cross-correlation function

$$r_{ij}(t) = E\{x_i(n)x_j(n+t)\} = E\{y_i(n)y_j(n+t)\}, \quad (1)$$

which is assumed to be well approximated (up to a scaling factor which will be ignored here) by the sample cross-correlation function (SCCF)

$$\hat{r}_{ij}(t) = \sum_{n=1}^{N} y_i(n)y_j(n+t) = y_i(t) * y_j(-t), \quad (2)$$

where the rightmost expression represents the SCCF by means of a convolution ($*$) operation. Note that in this expression, the time lag t takes the place of the time index n.

We define a first auxiliary signal by convolving the SCCF $\hat{r}_{ij}(t)$ with the microphone signal $y_j(t)$, and we again invoke the convolution notation to obtain the following equivalence,

$$\begin{aligned} s_{ijj}(t) &\triangleq \hat{r}_{ij}(t) * y_j(t) && (3) \\ &= y_i(t) * y_j(-t) * y_j(t) \\ &= y_i(t) * \hat{r}_{jj}(t), && (4) \end{aligned}$$

where $\hat{r}_{jj}(t)$ represents the sample autocorrelation function (SACF) of $y_j(t)$. Similarly, a second auxiliary signal is defined as follows,

$$\begin{aligned} s_{iij}(t) &\triangleq y_i(-t) * \hat{r}_{ij}(t) && (5) \\ &= y_i(-t) * y_i(t) * y_j(-t) \\ &= \hat{r}_{ii}(t) * y_j(-t), && (6) \end{aligned}$$

where $\hat{r}_{ii}(t)$ represents the SACF of $y_i(t)$. The auxiliary signals $s_{ijj}(t), s_{iij}(t)$ can be related to the so-called coskewness (third order central cross-moment) of the signals $y_i(t), y_j(t)$, but we will not explore this statistical interpretation. Instead, we consider the SACFs and SCCF in (3)-(6) as noncausal filters operating on the signals $y_i(t), y_j(t)$.

Consider the following sequence of signal processing and transmission operations:

[1]For ease of notation, the indices (i, j) will be used to denote the microphones as well as the nodes in which these microphones reside.

Step 1: The SACF $\hat{r}_{jj}(t)$ is computed in node j for lags $0 \leq t \leq \tau$ by using the entire length-N signal $y_j(n)$. The maximum lag $\tau \ll N$ should be chosen such that the computed samples $\hat{r}_{jj}(t)$ are representative for the entire SACF (which is possible whenever the SACF is a decaying or periodic function as is often the case for acoustic signals).
Step 2: The SACF samples $\hat{r}_{jj}(t), 0 \leq t \leq \tau$ are transmitted from node j to node i.
Step 3: The length-N auxiliary signal $s_{ijj}(t)$ is computed in node i by filtering the entire length-N signal $y_i(n)$ with the noncausal filter $\hat{r}_{jj}(t), -\tau \leq t \leq \tau$ having a symmetric impulse response constructed from the SACF samples $\hat{r}_{jj}(t), 0 \leq t \leq \tau$.
Step 4: A frame of $M \ll N$ samples is selected from the auxiliary signal $s_{ijj}(t)$ and is transmitted from node i to node j.
This 4-step sequence is also executed a second time, with interchanged indices i and j.

After executing the above procedure, the SCCF can be estimated by solving a deconvolution problem based on (3) in node j (making use of $s_{ijj}(t)$ and $y_j(t)$) as well as by solving a deconvolution problem based on (5) in node i (making use of $s_{iij}(t)$ and $y_i(t)$). If $M = N$, these deconvolution problems yield an exact estimate of the SCCF $\hat{r}_{ij}(t)$. In Section 3, we will propose three approaches and algorithms to solve these deconvolution problems in a distributed manner.

3. DISTRIBUTED CROSS-CORRELATION ESTIMATION ALGORITHMS

We first make the additional assumption that we are interested in estimating $\hat{r}_{ij}(t)$ only for time lags in a limited and known range $[\tau_1, \tau_2]$, using a similar motivation as in Step 1 in the above procedure, noting however that the range $[\tau_1, \tau_2]$ need not be centered around $t = 0$. If this range is unknown, it can be estimated by first applying the asymmetric approach outlined below for an extended range of time lags (since this is the only approach for which the amount of data to be transmitted among the nodes is independent of τ_1 and τ_2).

3.1. Symmetric deconvolution

Let us denote by $m+1$ the time index at which the length-M frame starts which is selected from the auxiliary signal $s_{ijj}(t)$ in Step 4 of the above procedure. Consider the $M \times (\tau_2 - \tau_1 + 1)$ Toeplitz and Hankel matrices

$$\mathbf{Y}_i = \begin{bmatrix} y_i(m+1-\tau_1) & \dots & y_i(m+1-\tau_2) \\ \vdots & \ddots & \vdots \\ y_i(m+M-\tau_1) & \dots & y_i(m+M-\tau_2) \end{bmatrix} \quad (7)$$

$$\tilde{\mathbf{Y}}_i = \begin{bmatrix} y_i(m+1+\tau_1) & \dots & y_i(m+1+\tau_2) \\ \vdots & \ddots & \vdots \\ y_i(m+M+\tau_1) & \dots & y_i(m+M+\tau_2) \end{bmatrix} \quad (8)$$

and $\mathbf{Y}_j, \tilde{\mathbf{Y}}_j$ defined similarly. Further consider the length-$(2\tau+1)$ SACF vector

$$\hat{\mathbf{r}}_{ii} = \begin{bmatrix} \hat{r}_{ii}(-\tau) & \dots & \hat{r}_{ii}(\tau) \end{bmatrix}^T \quad (9)$$

and $\hat{\mathbf{r}}_{jj}$ defined similarly, and the length-$(\tau_2 - \tau_1 + 1)$ SCCF vector to be estimated,

$$\hat{\mathbf{r}}_{ij} = \begin{bmatrix} \hat{r}_{ij}(\tau_1) & \dots & \hat{r}_{ij}(\tau_2) \end{bmatrix}^T . \quad (10)$$

Note that the computation of the length-M auxiliary signal frames in Step 3 and 4 of the above procedure can then be written as

$$\mathbf{s}_{ijj} = \mathbf{Y}_i \hat{\mathbf{r}}_{jj} \quad (11)$$

$$\mathbf{s}_{iij} = \tilde{\mathbf{Y}}_j \hat{\mathbf{r}}_{ii}. \quad (12)$$

The symmetric deconvolution approach consists in solving the following least-squares (LS) problem,

$$\hat{\mathbf{r}}_{ij,S} = \arg\min_{\hat{\mathbf{r}}_{ij}} \frac{1}{2} \left\| \begin{bmatrix} \mathbf{s}_{ijj} \\ \mathbf{s}_{iij} \end{bmatrix} - \begin{bmatrix} \mathbf{Y}_j \\ \tilde{\mathbf{Y}}_i \end{bmatrix} \hat{\mathbf{r}}_{ij} \right\|_2^2 , \quad (13)$$

in which the top M equations depend on data ($\mathbf{s}_{ijj}, \mathbf{Y}_j$) available in node j and the bottom M equations depend on data ($\mathbf{s}_{iij}, \tilde{\mathbf{Y}}_i$) available in node i. This LS problem can be solved in either of the two nodes, under the condition that $M+\tau_2-\tau_1$ data samples are additionally transmitted (either the elements of $\mathbf{Y}_j$ from node j to i or the elements of $\tilde{\mathbf{Y}}_i$ from node i to j).

3.2. Asymmetric deconvolution

With the aim of avoiding the additional transmission of $M + \tau_2 - \tau_1$ data samples as in the symmetric deconvolution approach, one could simply solve the upper half of the LS problem (13) in node j and the lower half in node i. This results in the asymmetric deconvolution approach, which yields two different SCCF estimates in the two nodes,

$$\hat{\mathbf{r}}_{ij,A}^{(j)} = \arg\min_{\hat{\mathbf{r}}_{ij}^{(j)}} \frac{1}{2} \left\| \mathbf{s}_{ijj} - \mathbf{Y}_j \hat{\mathbf{r}}_{ij}^{(j)} \right\|_2^2 \quad (14)$$

$$\hat{\mathbf{r}}_{ij,A}^{(i)} = \arg\min_{\hat{\mathbf{r}}_{ij}^{(i)}} \frac{1}{2} \left\| \mathbf{s}_{iij} - \tilde{\mathbf{Y}}_i \hat{\mathbf{r}}_{ij}^{(i)} \right\|_2^2 . \quad (15)$$

3.3. Consensus deconvolution

Whereas the symmetric deconvolution approach yields a unique SCCF estimate in both nodes at the cost of additional data transmission, the asymmetric deconvolution approach results in two different SCCF estimates but does not require additional data to be transmitted. A compromise between these two approaches can be found by reformulating the LS problem (13) as a consensus estimation problem, in which each node solves an asymmetric deconvolution problem subject to a consensus constraint. This consensus deconvolution approach hence requires the estimation of two local ($\hat{\mathbf{r}}_{ij}^{(i)}, \hat{\mathbf{r}}_{ij}^{(j)}$) and one consensus ($\hat{\mathbf{r}}_{ij}$) SCCF vector,

$$\min_{\hat{\mathbf{r}}_{ij}^{(i)}, \hat{\mathbf{r}}_{ij}^{(j)}, \hat{\mathbf{r}}_{ij}} \frac{1}{2} \left\| \mathbf{s}_{ijj} - \mathbf{Y}_j \hat{\mathbf{r}}_{ij}^{(j)} \right\|_2^2 + \frac{1}{2} \left\| \mathbf{s}_{iij} - \tilde{\mathbf{Y}}_i \hat{\mathbf{r}}_{ij}^{(i)} \right\|_2^2 \quad (16)$$

$$\text{s.t.} \quad \hat{\mathbf{r}}_{ij}^{(i)} = \hat{\mathbf{r}}_{ij} \quad (17)$$

$$\hat{\mathbf{r}}_{ij}^{(j)} = \hat{\mathbf{r}}_{ij}. \quad (18)$$

The alternating direction method of multipliers (ADMM) provides a suitable optimization framework for solving consensus problems in a distributed manner [6]. The ADMM requires the augmented Lagrangian

$$\begin{aligned} &\mathcal{L}_\rho \left(\hat{\mathbf{r}}_{ij}^{(i)}, \hat{\mathbf{r}}_{ij}^{(j)}, \hat{\mathbf{r}}_{ij}, \boldsymbol{\lambda}_{ij}^{(i)}, \boldsymbol{\lambda}_{ij}^{(j)} \right) = \frac{1}{2} \left\| \mathbf{s}_{ijj} - \mathbf{Y}_j \hat{\mathbf{r}}_{ij}^{(j)} \right\|_2^2 \\ &+ \frac{1}{2} \left\| \mathbf{s}_{iij} - \tilde{\mathbf{Y}}_i \hat{\mathbf{r}}_{ij}^{(i)} \right\|_2^2 + \boldsymbol{\lambda}_{ij}^{(j)T} (\hat{\mathbf{r}}_{ij}^{(j)} - \hat{\mathbf{r}}_{ij}) + \boldsymbol{\lambda}_{ij}^{(i)T} (\hat{\mathbf{r}}_{ij}^{(i)} - \hat{\mathbf{r}}_{ij}) \\ &+ \frac{\rho}{2} \left\| \hat{\mathbf{r}}_{ij}^{(j)} - \hat{\mathbf{r}}_{ij} \right\|_2^2 + \frac{\rho}{2} \left\| \hat{\mathbf{r}}_{ij}^{(i)} - \hat{\mathbf{r}}_{ij} \right\|_2^2 \end{aligned} \quad (19)$$

to be minimized sequentially w.r.t. the primal variables $\hat{\mathbf{r}}_{ij}^{(i)}, \hat{\mathbf{r}}_{ij}^{(j)}$, the consensus variables $\hat{\mathbf{r}}_{ij}$, and the dual variables $\boldsymbol{\lambda}_{ij}^{(i)}, \boldsymbol{\lambda}_{ij}^{(j)}$. The last two terms in (19) serve to smooth the Lagrangian in the neighborhood of the solution $\hat{\mathbf{r}}_{ij}^{(i)} = \hat{\mathbf{r}}_{ij}^{(j)} = \hat{\mathbf{r}}_{ij}$, and the parameter ρ plays a crucial role in the ADMM convergence behavior (see Section 4). Due to space limitations, the derivation of the ADMM for the problem under consideration is omitted, but the resulting algorithm is shown in Algorithm 1. Even though efficient stopping criteria for ADMM exist [6], we prefer to execute a fixed number of k iterations since the amount of data transmission required in Algorithm 1 scales linearly with k.

Algorithm 1 ADMM algorithm for consensus deconvolution

Input data vectors and matrices $\mathbf{s}_{ijj}, \mathbf{Y}_j$ (at node j) and $\mathbf{s}_{iij}, \tilde{\mathbf{Y}}_i$ (at node i), ADMM parameter ρ, initial variables $\hat{\mathbf{r}}_{ij}[0], \boldsymbol{\lambda}_{ij}^{(i)}[0]$ (at node j), and $\boldsymbol{\lambda}_{ij}^{(j)}[0]$ (at node i)

Output consensus SCCF estimate $\hat{\mathbf{r}}_{ij,C}$

at node j: $\boldsymbol{\Phi}_j = (\mathbf{Y}_j^T \mathbf{Y}_j + \rho \mathbf{I})^{-1}$, $\boldsymbol{\sigma}_j = \mathbf{Y}_j^T \mathbf{s}_{ijj}$

at node i: $\tilde{\boldsymbol{\Phi}}_i = (\tilde{\mathbf{Y}}_i^T \tilde{\mathbf{Y}}_i + \rho \mathbf{I})^{-1}$, $\boldsymbol{\sigma}_i = \tilde{\mathbf{Y}}_i^T \mathbf{s}_{iij}$

for $\kappa = 1, \ldots, k$ **do**

  at node j: $\hat{\mathbf{r}}_{ij}^{(j)}[\kappa] = \boldsymbol{\Phi}_j \left(\boldsymbol{\sigma}_j + \rho \hat{\mathbf{r}}_{ij}[\kappa-1] - \boldsymbol{\lambda}_{ij}^{(j)}[\kappa-1] \right)$

  at node i: $\hat{\mathbf{r}}_{ij}^{(i)}[\kappa] = \tilde{\boldsymbol{\Phi}}_i \left(\boldsymbol{\sigma}_i + \rho \hat{\mathbf{r}}_{ij}[\kappa-1] - \boldsymbol{\lambda}_{ij}^{(i)}[\kappa-1] \right)$

  transmit $\hat{\mathbf{r}}_{ij}^{(j)}[\kappa]$ from node j to i

  transmit $\hat{\mathbf{r}}_{ij}^{(i)}[\kappa]$ from node i to j

  at node j, i: $\hat{\mathbf{r}}_{ij}[\kappa] = \frac{1}{2} \left(\hat{\mathbf{r}}_{ij}^{(j)}[\kappa] + \hat{\mathbf{r}}_{ij}^{(i)}[\kappa] \right)$

  at node j: $\boldsymbol{\lambda}_{ij}^{(j)}[k] = \boldsymbol{\lambda}_{ij}^{(j)}[k-1] + \rho \left(\hat{\mathbf{r}}_{ij}^{(j)}[\kappa] - \hat{\mathbf{r}}_{ij}[\kappa] \right)$

  at node i: $\boldsymbol{\lambda}_{ij}^{(i)}[k] = \boldsymbol{\lambda}_{ij}^{(i)}[k-1] + \rho \left(\hat{\mathbf{r}}_{ij}^{(i)}[\kappa] - \hat{\mathbf{r}}_{ij}[\kappa] \right)$

end for

$\hat{\mathbf{r}}_{ij,C} = \hat{\mathbf{r}}_{ij}[k]$

4. SIMULATION RESULTS

The proposed approaches to distributed cross-correlation estimation are evaluated by means of Monte Carlo simulations. The microphone signals' correlated components $x_i(n), x_j(n)$ are generated by filtering a stationary Gaussian white noise signal of length $N = 2^{16}$ samples with two acoustic room impulse responses generated by means of the randomized image method [7] and truncated to 16 coefficients. The microphone signals $y_i(n), y_j(n)$ are obtained by adding Gaussian white measurement noise to $x_i(n), x_j(n)$. The time lag values are chosen as $\tau = -\tau_1 = \tau_2 = 20$. The estimation performance is measured by means of the normalized mean squared error,

$$\text{NMSE [dB]} = 10 \log_{10} \sum_{l=1}^{L} \frac{\|\hat{\mathbf{r}}_{ij,*} - \hat{\mathbf{r}}_{ij,0}\|_2^2}{\|\hat{\mathbf{r}}_{ij,0}\|_2^2} \quad (20)$$

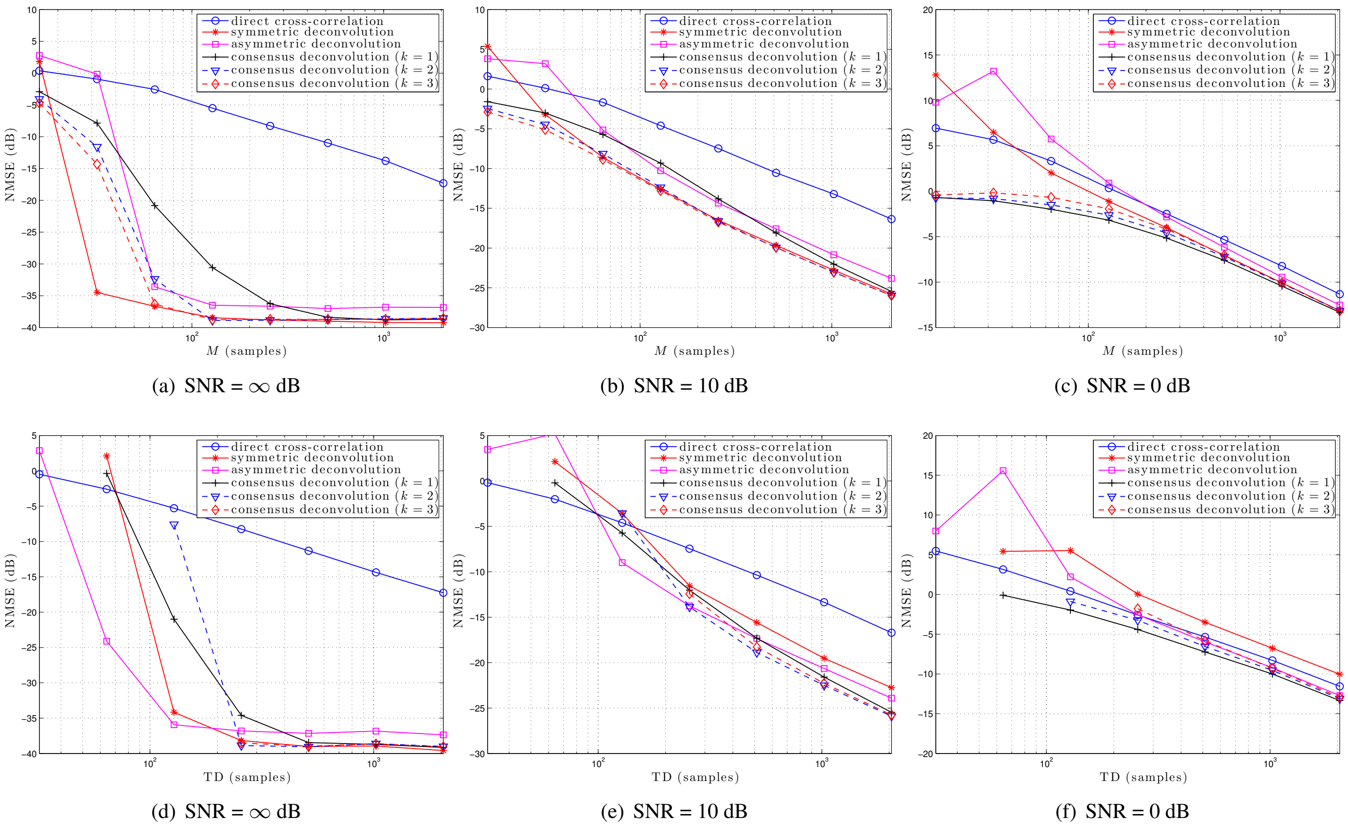
where the length-($\tau_2 - \tau_1 + 1$) SCCF vector $\hat{\mathbf{r}}_{ij,0}$ of the length-N noiseless signals $x_i(n), x_j(n)$ is used as the ground thruth, $\hat{\mathbf{r}}_{ij,*}$ represents the estimated SCCF vector obtained from either of the above approaches, and averaging is performed over $L = 100$ Monte Carlo trials. Matlab code for the different algorithms and for reproducing the simulation results and figures presented here are available online[2].

A first simulation, the results of which are not shown here due to space limitations, has been used to evaluate the influence of the ADMM parameter ρ for different values of the SNR and frame length M. It was observed that for small frame lengths ($M \leq 256$), the optimal choice for ρ depends on the SNR: in low-SNR conditions it is beneficial to give a larger weight to the last two terms in the augmented Lagrangian (19), thereby encouraging convergence to the consensus variable. In the next simulations we will use the values $\rho = 1, 0.2, 0.01$ for the cases SNR $= 0, 10, \infty$ dB.

In a second simulation, we compare the three deconvolution approaches to a direct computation of the SCCF based on the same amount of data M (i.e., using $N = M$ in (2)), see Fig. 1(a)–(c). From the results for SNR = ∞ dB, we can see that the deconvolution approaches suffer much less from finite-sample effects compared to the direct SCCF computation. Also, in noisy scenarios, the deconvolution approaches perform better. Overall, the asymmetric deconvolution approach performs slightly worse than the symmetric and consensus deconvolution approaches, and the performance differences fade out for increasing values of M. Finally, we observe that the consensus deconvolution ADMM algorithm provides a satisfactory result already after 1 iteration, and the benefit of executing 2 or 3 iterations is generally small.

In a third simulation the same algorithms are compared, but the value of M is chosen differently for each algorithm such that exactly the same amount of data needs to be trans-

[2]http://homes.esat.kuleuven.be/$\sim$tvanwate/software.html



(a) SNR = ∞ dB (b) SNR = 10 dB (c) SNR = 0 dB

(d) SNR = ∞ dB (e) SNR = 10 dB (f) SNR = 0 dB

Fig. 1. Comparison of direct and deconvolution approaches to cross-correlation estimation for: (a)–(c) variable truncation length M, (d)–(f) amount of transmitted data (TD).

Estimation approach	TD
direct cross-correlation	M
symmetric deconvolution	$2M + \tau + 1 + \tau_2 - \tau_1$
asymmetric deconvolution	$M + \tau + 1$
consensus deconvolution	$M + \tau + 1 + k(\tau_2 - \tau_1 + 1)$

Table 1. No. of transmitted data (TD) samples per node for different distributed cross-correlation estimation approaches

mitted among the nodes, as quantified by the number of transmitted data (TD) samples, see Table 1. The NMSE curves in Fig. 1(d)–(f) reveal that from this perspective, the symmetric deconvolution approach is less interesting compared to the asymmetric and consensus deconvolution approaches.

Summarizing, the presented simulation results illustrate that the consensus deconvolution approach with 1 or 2 ADMM iterations provides a suitable trade-off between the NMSE performance and the data transmission requirements.

REFERENCES

[1] A. Bertrand, S. Doclo, S. Gannot, N. Ono, and T. van Waterschoot, "Special issue on wireless acoustic sensor networks and ad hoc microphone arrays," *Signal Processing*, vol. 107, pp. 1–3, Feb. 2015.

[2] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks - Part I: sequential node updating," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5277–5291, Oct. 2010.

[3] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links – Part I: Distributed estimation of deterministic signals," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 350–364, Jan. 2008.

[4] Y. Zeng and R. C. Hendriks, "Distributed estimation of the inverse of the correlation matrix for privacy preserving beamforming," *Signal Processing*, vol. 107, pp. 109–122, Feb. 2015.

[5] 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, Jan. 2015.

[6] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[7] E. De Sena, N. Antonello, M. Moonen, and T. van Waterschoot, "On the modeling of rectangular geometries in room acoustic simulations," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 23, no. 4, pp. 774–786, Apr. 2015.