NETWORK TOPOLOGY SELECTION FOR DISTRIBUTED SPEECH ENHANCEMENT IN WIRELESS ACOUSTIC SENSOR NETWORKS

Joseph Szurley, Alexander Bertrand*, Ingrid Moerman‡ and Marc Moonen**

∗ Department of Electrical Engineering ESAT-SCD /
iMinds - Future Health Department, KU Leuven
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
E-mail: joseph.szurley@esat.kuleuven.be,
alexander.bertrand@esat.kuleuven.be,
marc.moonen@esat.kuleuven.be

‡ Department of Information Technology (INTEC)
iMinds - Ghent University
Gaston Crommenlaan 8, Bus 201, 9050 Ghent, Belgium
E-mail: ingrid.moerman@intec.ugent.be

ABSTRACT

A wireless acoustic sensor network is envisaged where each node estimates a locally observed speech signal that has been corrupted by additive noise. The nodes perform noise reduction by means of the distributed adaptive node-specific signal estimation algorithm in a tree topology (T-DANSE). The T-DANSE algorithm inherently relies on a network that has been pruned to a tree topology where a single node has been designated as the *root* node. We will demonstrate that, due to the data-driven flow of the T-DANSE algorithm and unavoidable errors in the estimation of certain second-order statistics, the selection of the root node and the pruning of an ad-hoc network to a tree topology play an important role in the overall performance of the speech enhancement algorithm in terms of noise reduction as well as input-output delay. With this in mind we introduce the concept of eigenvector centrality with a weighted adjacency matrix that can be used to select a root node, as well as to prune an ad-hoc network to a specific tree topology that yields good speech enhancement performance when applying the T-DANSE algorithm.

Index Terms— Distributed signal estimation, wireless acoustic sensor networks, network topology, eigenvector centrality

1. INTRODUCTION

Acoustic noise reduction techniques aim to estimate a desired speech signal that has been corrupted by noise. While single microphone techniques may introduce speech distortion in order to reduce the noise, these effects can be mitigated by the use of multi-microphone techniques.

Multi-microphone noise reduction algorithms are able to overcome the limitations of single microphone techniques not only from the use of added channels (auxiliary microphones) but also because they are able to capture increased spatial information due to the spacing of the microphones [1, 2].

In order to further increase the spatial diversity of the information collected at the microphones, noise reduction algorithms may be implemented to work with multiple devices, or nodes, that each contain a set of microphones and are distributed throughout the sensing environment. This type of spatially distributed microphone network creates what is referred to as a wireless acoustic sensor network (WASN).

In this paper we envisage a WASN where each node estimates a locally observed speech signal that has been corrupted by noise. It performs this estimation by means of the distributed adaptive node-specific signal estimation algorithm in a tree topology (T-DANSE), where each node linearly combines (i.e. fuses) its own signals with signals from neighboring nodes before forwarding them to the next node. The T-DANSE algorithm was first introduced in [3], however a specific application in terms of a WASN has not been explored.

In the previous implementation of the T-DANSE algorithm it was assumed that the network had been pruned to that of a spanning tree. However due to the so-called *data-driven* flow in the T-DANSE algorithm, the way that the tree topology is formed, especially the selection of the *root* node, is shown to have a significant impact on the performance of the T-DANSE algorithm. This is mainly due to the fact that there are estimation errors in certain second-order statistics estimates at each node which tend to become more prevalent with decreasing input signal-to-noise ratio (SNR_{IN}).

Since all signals have to pass through the root node, it is important that the signal fusion rule at this node is correctly estimated, which is only possible if the SNR_{IN} at the root node is sufficiently high. A similar statement (be it to a lesser de-

This research 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 P7/19 'Dynamical systems, control and optimization' (DYSCO) 2012-2017 and Research Project iMinds, Research Project FWO nr. G.0763.12 'Wireless Acoustic Sensor Networks for Extended Auditory Communication'. Alexander Bertrand is supported by a Postdoctoral Fellowship of the Research Foundation Flanders (FWO). The scientific responsibility is assumed by its authors.

gree) holds for the direct neighbors of the root node, as well as their respective neighbors, etc. Furthermore, to minimize the input-output delay within the network, the root node should be close (in terms of number of hops) to any other node in the network.

Before the T-DANSE algorithm begins we assume that the network initially has an ad-hoc topology where each node only communicates with nearby nodes. We introduce the concept of eigenvector centrality with an SNR_{IN} weighted adjacency matrix which will allow us to identify a suitable root node and to prune the network to a tree topology, which satisfies the earlier mentioned properties. A particularly convenient property of the eigenvector centrality is that it can be computed in a distributed fashion.

The paper is organized as follows: In Section 2 the T-DANSE algorithm is reviewed along with its data-driven flow. The estimation of signal statistics is given in Section 3 and what impact these along with root node selection have on the performance of the T-DANSE algorithm. In Section 4 a weighted eigenvector centrality measure is introduced which is used in pruning an ad-hoc network to a tree topology as well as root node selection. Section 5 contains simulation results and conclusions are given in Section 6.

2. T-DANSE ALGORITHM

A WASN is envisaged that contains J nodes with node k having $M_k,\ k \in 1 \dots J$ microphone signals. We assume that the network has a tree topology, i.e., it cannot contain any loops. The received microphone signals, at node k, can be represented in the short-time Fourier transform (STFT) domain at a given frequency ω and time t as

$$y_{k,m}(\omega, t) = x_{k,m}(\omega, t) + n_{k,m}(\omega, t),\ m = 1 \dots M_k \quad (1)$$

where x is the desired speech component and n is an additive uncorrelated noise component. For ease of exposition the frequency and time indices will be omitted bearing in mind that the following operations take place in the STFT domain.

2.1. T-DANSE Basic Operation

In the T-DANSE algorithm the goal of each node in the WASN is to estimate its own node specific desired speech signal, d_k, which, without loss of generality, is assumed to be the speech component in the first microphone of the node, $d_k = x_{k,1}$. It accomplishes this estimation by applying a linear filter that fuses all of its input signals into a single output signal that serves as an estimate for d_k. The set of input signals consists of the node's own M_k microphone signals as well as (fused) signals obtained from neighboring nodes (see further). In order to propagate information throughout the network, the node also broadcasts this output signal to other nodes in its neighborhood.

The remarkable aspect of the T-DANSE algorithm is that the linear fusion rules at the different nodes converge to an equilibrium setting that yield node-specific signal estimates identical to those obtained when each node would have access to all the microphone signals in the network[1]. We briefly outline the T-DANSE algorithm and the reader is referred to [3] for a more in depth discussion as well as convergence proofs. Note that a full understanding of the T-DANSE is not intended and only key concepts will be outlined.

We denote the neighbors, or the set of nodes that are connected to node k excluding node k, as N_k and the broadcast signals from these nodes as $z_q,\ \forall q \in N_k$, which are contained in a stacked vector $\mathbf{z}_{k_{-k}}$ where the $-k$ indicates that node k's z_k signal is not included. We will define these z_q signals in the sequel (see equations (5) and (6)).

Instead of decompressing the received signals from other nodes, node k applies a scaling parameter to each element of $\mathbf{z}_{k_{-k}}$ defined as $g_{kq},\ \forall q \in N_k$, which are contained in a stacked vector $\mathbf{g}_{k_{-k}}$.

Each node updates its node-specific parameters per node, $\mathbf{w}_{kk}$ and $\mathbf{g}_{k_{-k}}$, in an iterative fashion by solving the local node-specific linear minimum mean square error problem,

$$\begin{bmatrix} \mathbf{w}_{kk}^{i+1} \\ \mathbf{g}_{k_{-k}}^{i+1} \end{bmatrix} = \underset{\mathbf{w}_{kk}, \mathbf{g}_{k_{-k}}}{\arg\min}\ E\left\{ \left| d_k - \begin{bmatrix} \mathbf{w}_{kk} \\ \mathbf{g}_{k_{-k}} \end{bmatrix}^H \begin{bmatrix} \mathbf{y}_k \\ \mathbf{z}_{k_{-k}} \end{bmatrix} \right|^2 \right\} \quad (2)$$

where $E\{.\}$ denotes the expectation operator, the superscript i denotes the iteration index and H is the Hermitian transpose. For ease of exposition we denote $\tilde{\mathbf{y}}_k = [\mathbf{y}_k^T\ \mathbf{z}_{k_{-k}}^T]^T$ where $\tilde{\mathbf{x}}_k$ is defined similarly and so contains the desired speech components of $\tilde{\mathbf{y}}_k$. The solution to (2) is given as the multi-channel Wiener filter (MWF)

$$\begin{bmatrix} \mathbf{w}_{kk}^{i+1} \\ \mathbf{g}_{k_{-k}}^{i+1} \end{bmatrix} = \mathbf{R}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k}^{-1} \mathbf{R}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k} \tilde{\mathbf{e}}_k \quad (3)$$

where $\mathbf{R}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k} = E\{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k^H\}$, $\mathbf{R}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k} = E\{\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^H\}$, and $\tilde{\mathbf{e}}_k$ is a vector with the first entry equal to 1 and all other equal to 0, which selects the first column of $\mathbf{R}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k}$. We will discuss in Section 3 how to estimate $\mathbf{R}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k}$ and $\mathbf{R}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k}$. The estimated desired signal at each node is then given as (cfr. (2))

$$\overline{d_k} = (\mathbf{w}_{kk}^{i+1})^H \mathbf{y}_k + (\mathbf{g}_{k_{-k}}^{i+1})^H \mathbf{z}_{k_{-k}}. \quad (4)$$

In the T-DANSE algorithm it can be shown that the MWF (3) and the estimated desired signal (4) at each node converge to the same solution as if the node had access to all of the microphone signals in the WASN [3]. In Section 3 we will discuss how estimation errors affect the convergence of T-DANSE algorithm and in Section 4 how to use this information to reduce the effect of these estimation errors through proper pruning of the initial ad-hoc network to a tree topology.

[1] It is noted that in theory the T-DANSE algorithm converges to the same solution as if the nodes have access to all microphones in the network. However in practice, due to errors in the estimation of the signal statistics as well as non-stationarities, differences between the two solutions normally occur.

2.2. Data-driven flow in T-DANSE

In order to pass information throughout the network we first assume that the broadcast signal from node k is given by the following fusion rule

$$z_k = \mathbf{w}_{kk}^H \mathbf{y}_k + \sum_{q \in N_k} g_{kq}^* z_q. \tag{5}$$

where $*$ indicates the complex conjugate. In using (5), however, it was shown in [3] that indirect feedback becomes a problem in the WASN. If the WASN contains feedback, i.e., the signals node k receives from its neighbors contain contributions from node k's own signals, then the T-DANSE algorithm is unable to converge to the optimal solution.

In order to avoid feedback the nodes can transmit what is referred to as transmitter feedback cancellation (TFC) signals where it is assumed that each node pair has a reserved point-to-point communication link. We define the signal which is transmitted from node k to node q as

$$\begin{aligned} z_{kq} &= \mathbf{w}_{kk}^H \mathbf{y}_k + \sum_{l \in N_k \setminus \{q\}} g_{kl}^* z_{lk} \\ &= z_k - g_{kq}^* z_{qk}. \end{aligned} \tag{6}$$

where z_{qk} is the signal which is transmitted from node q to node k. Note that z_{kq} consists of a linear combination of the microphone signals of node k and the z-signals obtained from its neighbors with z_{qk} excluded.

It is readily apparent that the transmitted signal from node k to node q relies on the transmitted signal from node q to node k. In order to rectify this dead-lock we discuss how the T-DANSE algorithm can intuitively remove this by means of backward substitution from the data-driven flow in the WASN.

2.3. Fusion Flow

The fusion flow is initiated at the leaf nodes, i.e., nodes with a single neighbor. Every time a new STFT frame can be computed from the new microphone signal observations, the leaf nodes fuse their local microphone signals by means of (6). Note that since a leaf node has a single neighbor (6) reduces to $z_k = \mathbf{w}_{kk}^H \mathbf{y}_k$.

Once a non-leaf node has received all of the signals from its neighbors (except for one) it fuses its microphone signals with its received signals by means of (6). This continues until all of the information arrives at the most central node, which is referred to as the root node.

2.4. Diffusion Flow

The diffusion flow is initiated once all of the fusion flow signals have reached the root node. The root node initiates the diffusion flow by transmitting the TFC-signals as given in (6) to each of its neighbors. This continues through the neighboring nodes until the data is spread out through the entire network, i.e., ending at the leaf nodes.

3. ESTIMATION OF SECOND-ORDER STATISTICS

The MWF (3) calculated at each node not only relies on the perfect estimation of the second-order statistics of the received signal, $\mathbf{R}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k}$, but also of the unobservable desired speech signal, $\mathbf{R}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k}$. In practice, however, the second-order statistics can only be imperfectly estimated by collected observations denoted as $\tilde{\mathbf{R}}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k}$ and $\tilde{\mathbf{R}}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k}$. For this estimation we rely on the fact that speech has an on-off behavior which is discerned by a so-called voice activity detector (VAD).

The VAD is considered active during segments when speech+noise is present in which the so-called speech+noise correlation matrix can be estimated in a recursive fashion by means of an exponential forgetting factor, i.e.,

$$\tilde{\mathbf{R}}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k}[t] = \lambda \tilde{\mathbf{R}}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k}[t-1] + (1-\lambda)\tilde{\mathbf{y}}_k[t]\tilde{\mathbf{y}}_k[t]^H \tag{7}$$

where $0 < \lambda < 1$. The noise correlation matrix $\tilde{\mathbf{R}}_{\tilde{\mathbf{n}}_k\tilde{\mathbf{n}}_k}$ is estimated in a similar fashion during noise only frames when the VAD is considered inactive. Since it is assumed that the desired speech and noise are uncorrelated the second-order statistics of the desired speech signal may be estimated by

$$\tilde{\mathbf{R}}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k} = \tilde{\mathbf{R}}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k} - \tilde{\mathbf{R}}_{\tilde{\mathbf{n}}_k\tilde{\mathbf{n}}_k}. \tag{8}$$

Notice that when a node k has low SNR_{IN} microphone signals the desired speech signal correlation matrix is poorly estimated resulting in large relative errors on the entries of $\tilde{\mathbf{R}}_{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k}$.

In [4] it was shown that when using imperfectly estimated correlation matrices and, if the errors are considered small with respect to $\mathbf{R}_{\tilde{\mathbf{n}}_k\tilde{\mathbf{n}}_k}$, the MWF becomes the optimal filter given by (3) and an extra bias term. We therefore assume that nodes which contain microphones with larger SNR_{IN} tend to have smaller estimation errors, which leads to smaller errors in the optimal filter (3).

We see that, due to the date-driven flow of the T-DANSE algorithm, if a root node is chosen with a low SNR_{IN} this not only places a bias term on its own estimation but also on the TFC-signals that the root node transmits to its neighbors. This in turn effects the estimation within those nodes, and their respective neighbors, etc., i.e., the error ripples throughout the whole network. Therefore since all data is fused through the root node it is important to have a good estimation of the signal statistics in the root node in order to decrease the added bias term as much as possible.

It is with these low SNR_{IN} nodes and the data-driven flow of T-DANSE in mind that we look to not only pick a root node with a high SNR_{IN}, thereby reducing the errors spread throughout the network, but also construct the entire tree topology based around this idea.

4. WEIGHTED EIGENVECTOR CENTRALITY

In Section 3 it was assumed that nodes with low SNR_{IN} tend to have larger estimation errors in the estimated correlation matrices which increases the bias term on their calculated filters. We see that in the T-DANSE algorithm this effect is compounded due to the fact that the signals are linearly compressed with these imperfect filters and fused with signals from other nodes. We therefore use the assumption that nodes with high SNR_{IN} have lower estimation errors to help prune an ad-hoc network to a tree topology while trying to keep the highest SNR_{IN} toward the root node and nodes with low SNR_{IN} toward the leaf nodes.

We first assume, that before the commencement of the T-DANSE algorithm, the network is connected in an ad-hoc fashion, i.e., nodes only communicate with the neighbors that fall within their transmission range. We define the so-called adjacency matrix $\mathbf{A}$ from the original ad-hoc topology as

$$a_{kq} = a_{qk} = \begin{cases} 1 & \text{if } k \text{ is connected to } q \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

which categorizes the links of the individual nodes.

Our first aim is to obtain a small input-output delay, i.e., the maximum distance between two nodes (in number of hops) should be as small as possible. This means that the tree should be highly branched, i.e., we need to find a root node which has many neighbors who - at their turn - also have many neighbors, etc. An interesting approach to identify such a potential root node is based on the so-called eigenvector centrality, which calculates the centrality of each node [5]. It is based on the principle that a node should get a high eigenvector centrality if it has many neighbors who - at their turn - also have a high eigenvector centrality. This leads to the following definition for the eigenvector centrality of node k [5]

$$c_k = \frac{1}{\alpha} \sum_{q \in N_k} c_q, \ \forall k \in J \quad (10)$$

where α is an arbitrary normalization factor, which can be re-written in terms of the adjacency matrix $\mathbf{A}$ as [5]

$$\alpha \mathbf{c} = \mathbf{A}\mathbf{c}. \quad (11)$$

It can be shown that if only non-negative coefficients are allowed in c then there is a unique solution to (10) given by the principle eigenvector of $\mathbf{A}$ that corresponds to the largest eigenvalue α_{max}. It is important to note that this principle eigenvector can easily be computed in a distributed fashion (see [5]).

A root node is then selected which corresponds to the node with the highest eigenvector centrality. The ad-hoc network may then be pruned to a tree topology by connecting nodes based on their eigenvector centrality, i.e., after the root node is selected the node with the next highest eigenvector centrality is connected, if a link exists, where the process continues until all nodes that were available in the ad-hoc network are included in the tree topology. Note that there is no assumption on the maximum number of connections per node, but this can easily be included as a constraint in the algorithm.

However, in terms of the estimation problem at hand, the eigenvector centrality of a node may not lead to a good choice for the root node, e.g., a node may have a low SNR_{IN} and have many connections with other nodes which, if chosen as the root node, will affect the performance of the T-DANSE algorithm.

We therefore propose the use of a weighted adjacency matrix to not only take into account the eigenvector centrality of a node but also use the SNR_{IN} in order to select a good root node. This will also be used to prune the original ad-hoc network to a tree topology. Similar weighting strategies have been studied in [6] which rely only on the degree centrality of a node and SNR_{IN} was used for designing topologies in [7].

We first assume that the microphones of a node have a similar SNR_{IN}, where the SNR_{IN} at node k is denoted as $\text{SNR}_{\text{IN},k}$ and is defined as the ratio of signal power to noise power. We define a new weighted eigenvector centrality based on the following principle. A node should receive a high weighted eigenvector centrality if it has many neighbors with a high-SNR_{IN}, who also have many neighbors with high-SNR_{IN}, etc. Furthermore each node's weighted eigenvector centrality should also be weighted with its own SNR_{IN} in order to avoid that a low-SNR_{IN} node surrounded by high-SNR_{IN} nodes be given a high weighted eigenvector centrality. This leads to the following definition of a weighted eigenvector centrality,

$$c_k = \frac{1}{\alpha} \text{SNR}_{\text{IN},k} \sum_{q \in N_k} c_q, \ \forall k \in J. \quad (12)$$

This weighted eigenvector centrality can again be shown to have a unique solution given by the principle eigenvector of a weighted adjacency matrix $\mathbf{A}_{\text{SNR}}$ [5]. In order to construct the weighted adjacency matrix, $\mathbf{A}_{\text{SNR}}$, corresponding to (12), a diagonal matrix, $\mathcal{D}_{\text{SNR}_{\text{IN}}}$, is constructed where the diagonal entries are equal to the SNR_{IN} of the nodes[2]

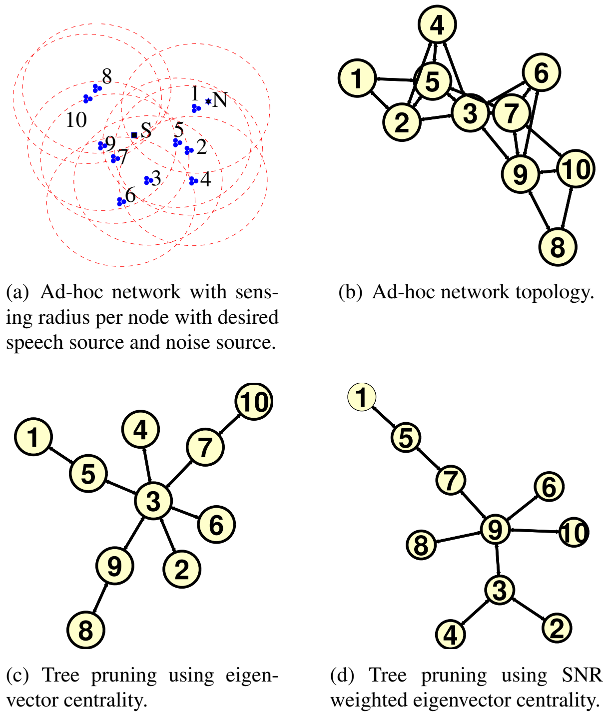
$$\mathcal{D}_{\text{SNR}_{\text{IN}}} = \begin{bmatrix} \text{SNR}_{\text{IN},1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \text{SNR}_{\text{IN},J} \end{bmatrix}. \quad (13)$$

The weighted adjacency matrix can then be given as

$$\mathbf{A}_{\text{SNR}} = \mathcal{D}_{\text{SNR}_{\text{IN}}} \mathbf{A}. \quad (14)$$

The root node is selected as the node with the highest weighted eigenvector centrality found by replacing $\mathbf{A}$ by $\mathbf{A}_{\text{SNR}}$ in (10), which can again be easily computed in a distributed fashion. The tree topology is then formed in a similar fashion where nodes are connected based on their weighted eigenvector centrality.

[2]Since only non-negative coefficients guarantee a unique solution to (12) the SNR_{IN} is given as the ratio of signal power to noise power and not the log of the ratio (dB).



(a) Ad-hoc network with sensing radius per node with desired speech source and noise source.

(b) Ad-hoc network topology.

(c) Tree pruning using eigenvector centrality.

(d) Tree pruning using SNR weighted eigenvector centrality.

Fig. 1. Original sensing environment and network topologies.

5. SIMULATIONS

We assume that the ad-hoc network is initially formed as in Figure 1(a) where there are 10 nodes each with a sensing radius of 2.5m. The initial topology of the ad-hoc network is given in Figure 1(b). Each node is assumed to have 3 microphones that are placed equidistantly around the center of the node at a radius of 1 cm. A single target speech source ■ (denoted by S) is present together with an additive white noise source $*$ (denoted by N). Uncorrelated white noise that is 10% of the average power of the noise source is added to each microphone which is representative of sensor noise.

The room dimensions are 5x5x5m where a reflection coefficient of 0.2 is used for all surfaces. A STFT block length of L = 128 is used with a sampling frequency of $f_s = 8000$ Hz. The estimation of the speech+noise and noise correlation matrices uses an ideal VAD.

Before starting the T-DANSE algorithm the root node is selected as the node with either the highest eigenvector centrality (10) or SNR_{IN} weighted eigenvector centrality (12). The values for the different eigenvector centrality measures are given in Table 1 along with the corresponding SNR_{IN} of each reference microphone and the nodes selected as the root node are in bold. While the SNR_{IN} of the reference microphones is given in dB in Table 1, the values used in the weighted adjacency matrix (14) are not in dB as explained in Section 4. The ad-hoc network is then pruned to a tree, Figure 1(c) and Figure 1(d), by using the method given in Section 4.

Node	SNR_{IN}	E.C.	SNR_{OUT}	W.E.C.	SNR_{OUT}
1	-14.63	0.161	3.26	0.001	3.64
2	-3.32	0.303	7.63	0.027	8.65
3	0.96	**0.473**	10.44	0.282	10.67
4	-3.33	0.273	6.15	0.027	7.23
5	-1.25	0.381	8.72	0.102	10.63
6	-0.84	0.291	5.53	0.180	7.70
7	5.16	0.403	12.68	0.641	13.43
8	-0.82	0.133	7.59	0.097	8.22
9	5.24	0.356	11.20	**0.648**	13.01
10	0.02	0.210	8.65	0.194	8.77

Table 1. SNR_{IN} (dB), eigenvector centrality (E.C.), SNR_{OUT} (dB) using E.C., SNR_{IN} weighted eigenvector centrality (W.E.C.) and SNR_{OUT} (dB) using SNR_{IN} W.E.C. for each node. Root nodes for each centrality measure are given in bold.

Notice that in using the weighted eigenvector centrality to prune the ad-hoc network the output signal-to-noise ratio (SNR_{OUT}) performance on all the nodes increases. In using the weighted eigenvector centrality the root node was selected as the node with the highest SNR_{IN} which also has higher SNR_{IN} nodes connected to it.

6. CONCLUSIONS

The T-DANSE algorithm was presented for use in a WASN. Due to the errors in the estimation of the second-order signal statistics it was shown that the selection of the root node is of paramount importance. The method of using a so-called weighted eigenvector centrality was presented to not only select a good root node for the network but to also prune an already existing ad-hoc network to a tree topology. Simulations showed that using a weighted eigenvector centrality compared to that of an unweighted eigenvector centrality not only changed the pruned network topology but also improved the performance of the T-DANSE algorithm.

7. REFERENCES

[1] J. Chen, J. Benesty, Y.A. Huang, and S. Doclo, "New insights into the noise reduction wiener filter," *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 14, no. 4, pp. 1218 –1234, Jul. 2006.

[2] T. Lotter and P. Vary, "Dual-channel speech enhancement by superdirective beamforming," *EURASIP J. Adv. Sig. Proc.*, 2006.

[3] A. Bertrand and M. Moonen, "Distributed adaptive estimation of node-specific signals in wireless sensor networks with a tree topology," *IEEE Trans. on Signal Process.*, vol. 59, no. 5, pp. 2196–2210, May 2011.

[4] B. Cornelis, M. Moonen, and J. Wouters, "Performance analysis of multichannel Wiener filter-based noise reduction in hearing aids under second order statistics estimation errors," *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 19, no. 5, pp. 1368–1381, Jul. 2011.

[5] A. Bertrand and M. Moonen, "Seeing the bigger picture: How nodes can learn their place within a complex ad hoc network topology," *IEEE Signal Processing Magazine*, 2013.

[6] T. Opsahl, F. Agneessens, and J. Skvoretz, "Node centrality in weighted networks: Generalizing degree and shortest paths," *Social Networks*, vol. 32, no. 3, pp. 245 – 251, 2010.

[7] S. Kar and J.M.F. Moura, "Sensor networks with random links: Topology design for distributed consensus," *CoRR*, vol. abs/0704.0954, 2007.