Learning-based Acoustic Source Localization in Acoustic Sensor Networks using the Coherent-to-Diffuse Power Ratio

Andreas Brendel and Walter Kellermann
Multimedia Communications and Signal Processing, Friedrich-Alexander-Universität Erlangen-Nürnberg,
Cauerstr. 7, D-91058 Erlangen, Germany
{Andreas.Brendel, Walter.Kellermann}@FAU.de

***Abstract*—A distributed learning-based algorithm for the localization of acoustic sources in an acoustic sensor network is proposed. It is based on estimates of the Coherent-to-Diffuse Power Ratio (CDR), which serve as feature for the source-microphone distance, i.e., the range. The relation between the estimated CDR and the range is learned by using Gaussian processes for non-parametric regression. The range estimates obtained from evaluating the regression function are fused by a weighted least squares estimation, which is implemented recursively, allowing for a distributed version of the algorithm. The resulting method is computationally efficient, works in highly reverberant and noisy scenarios and needs only a small amount of data shared over the network. The training phase of the algorithm requires only a few labeled observations. We show the efficacy of the approach with data obtained from image-source simulation.**

***Index Terms*—Coherent-to-Diffuse Power Ratio, Gaussian Process Regression, Weighted Least Squares, Distributed Algorithm, Acoustic Sensor Network, Localization**

I. Introduction

Acoustic Sensor Networks (ASNs) allow to increase the coverage of an area of interest by spatial distribution of the sensors and according research gained popularity in the signal processing community [1]. Especially the localization and tracking of acoustic sources attracted considerable research efforts [2]. Several approaches have been proposed which mainly rely on estimates of the Direction of Arrival (DOA) or the Time Difference of Arrival (TDOA) of the sources of interest, e.g., triangulation of narrowband DOA estimates [3], clustering of phase differences between observed signals by the EM algorithm [4] and a distributed version of this algorithm in [5]. The performance of DOA-based methods usually degrades significantly in highly reverberant scenarios [6], and in distributed sensor networks they suffer from the problem of ghost sources [7], i.e., false combinations of DOA estimates. Another group of algorithms estimates the range of the acoustic source, e.g., by the observed signal energy at the sensor nodes [8]. The obtained range estimates can be fused by a Weighted Least Squares (WLS) estimate [9], which can be implemented in a distributed fashion [10]. However, energy-based localization methods such as [8] assume a free-field propagation model, which is not applicable in enclosures due to reverberation. The estimation of the range of an acoustic source is a difficult problem which has been tackled by approaches based on prior knowledge on the Room Impulse Responses (RIRs) [11], [12] or physical parameters characterizing the room [13]. However, this knowledge is usually not available in practice. Therefore, another class of algorithms has been developed which avoids the need of this prior knowledge by a learning phase [14]–[17]. The characteristic properties of a diffuse sound field can be used to infer distance, e.g., for calibration of ASNs [18]. The CDR [19], which is the power ratio of the direct and the diffuse signal components, can be used as a feature for the range of the acoustic source. However, due to the lack of knowledge about the room characteristics, the relation between range and CDR is unknown in practice. Standard regression approaches like polynomial regression etc. are not applicable here, because the fitting function is also unknown. Therefore, we choose a non-parametric approach for regression, based on Gaussian Processes (GPs), in this paper. In other disciplines, arising from geostatistics, GP regression is also known as Kriging [20] and it was used in sensor networks, e.g., for extending the coverage of the area of interest by interpolation of sensor observations [21].

In this contribution, we propose a distributed acoustic source localization scheme based on range estimation using GP regression to learn the relation between the range of the source and the estimated CDR. The presented approach for localization using the range estimates is formulated as the solution of a WLS problem and designed in a distributed fashion, which allows to distribute the computational load over the network and to produce instantaneous estimates of the source position based on current observations. The computational complexity of the algorithm as well as the amount of necessary data transfer between the nodes is very low.

II. Range Estimation

The first part of the algorithm is the collection of training data and the range estimation by GP regression.

A. Feature Calculation

We assume a set of M sensor nodes, each equipped with two microphones with spacing $d_{\text{mic},m}$, distributed over the area

This work was supported by DFG under contract no <Ke890/10-1> within the Research Unit FOR2457 ”Acoustic Sensor Networks”.

$$\widehat{\mathrm{CDR}}_m = \frac{\Gamma_n^{(m)}\,\mathrm{Re}\left\{\hat{\Gamma}_x^{(m)}\right\} - \left|\hat{\Gamma}_x^{(m)}\right|^2 - \sqrt{\left(\Gamma_n^{(m)}\right)^2 \mathrm{Re}\left\{\hat{\Gamma}_x^{(m)}\right\}^2 - \left(\Gamma_n^{(m)}\right)^2 \left|\hat{\Gamma}_x^{(m)}\right|^2 + \left(\Gamma_n^{(m)}\right)^2 - 2\,\Gamma_n^{(m)}\,\mathrm{Re}\left\{\hat{\Gamma}_x^{(m)}\right\} + \left|\hat{\Gamma}_x^{(m)}\right|^2}}{\left|\hat{\Gamma}_x^{(m)}\right|^2 - 1} \quad (1)$$

of interest. The microphone signal $x_{i,m}(t)$ at sensor node m is modeled as the superposition of the anechoic speech signal $s_{i,m}(t)$ and an additional reverberation/noise signal $n_{i,m}(t)$

$$x_{i,m}(t) = s_{i,m}(t) + n_{i,m}(t), \quad i \in \{1, 2\}. \quad (2)$$

The auto/cross Power Spectral Density (PSD) of the microphone signals can be estimated by recursive averaging over time using a forgetting factor λ

$$\hat{\Phi}_{x_i x_j}^{(m)}(l, f) = \lambda \hat{\Phi}_{x_i x_j}^{(m)}(l-1, f) + (1-\lambda) X_{i,m}(l, f) X_{j,m}^*(l, f),$$

where $i, j \in \{1, 2\}$ and $X_{i,m}$, $X_{j,m}$ are the Short-Time Fourier Transform (STFT) domain representations of the signals for time frame l and frequency f, observed at microphone i, j, respectively. The complex spatial coherence function of the observed signals is estimated as

$$\hat{\Gamma}_x^{(m)}(l, f) = \frac{\hat{\Phi}_{x_1 x_2}^{(m)}(l, f)}{\sqrt{\hat{\Phi}_{x_1 x_1}^{(m)}(l, f) \hat{\Phi}_{x_2 x_2}^{(m)}(l, f)}}. \quad (3)$$

Here, the DOA-independent CDR estimator (1) proposed in [19] is employed, where the dependency on time frame l and frequency f is discarded in (1) for a concise notation. The coherence of a diffuse sound field is given by

$$\Gamma_n^{(m)}(f) = \frac{\sin(2\pi f d_{\mathrm{mic},m}/c)}{2\pi f d_{\mathrm{mic},m}/c} \quad (4)$$

with c as the speed of sound. As the feature of the regression model, we define the averaged diffuseness

$$\widehat{\gamma}_m = \frac{1}{N_t(f_{\max} - f_{\min} + 1)} \sum_{l=1}^{N_t} \sum_{f=f_{\min}}^{f_{\max}} \frac{1}{\widehat{\mathrm{CDR}}_m(l, f) + 1}, \quad (5)$$

where N_t denotes the number of time frames, and $f_{\min}$ and $f_{\max}$ defines the minimum and maximum considered frequency, respectively. Note that $\widehat{\gamma}_m \in [0, 1]$ follows from the definition of the feature (5).

B. Non-parametric Regression

The relation between the averaged diffuseness $\widehat{\gamma}_m$ and the range is unknown because the characteristics of the room are not accessible in general. Therefore, we aim at estimating a regression function to learn this relation. Since we do not even know the general shape of the regression function, we use GP regression [22], because this non-parametric approach does not require assumptions about the class of a parametric regression function.

To discriminate between the training and the localization data, we mark the training data with a tilde $\tilde{(\cdot)}$. The nodes compute the averaged diffuseness $\tilde{\gamma}_m$ in the training phase, equipped with the label of the correct range between source and node. These labeled training data points are shared between the nodes. We assume that, at the current node m, we have received and calculated N_{train} labeled training points, which are stacked in a vector $\tilde{\boldsymbol{\gamma}}_m \in [0, 1]^{N_{\text{train}}}$. The corresponding ranges are stacked in the vector $\tilde{\mathbf{r}}_m \in \mathbb{R}_+^{N_{\text{train}}}$. The single training data points are indexed with i and j.

We model the range $\tilde{r}_{m,i}$ of training point i available at node m to be related to the corresponding $\tilde{\gamma}_{m,i}$ by a function f and to be corrupted by Gaussian noise of zero mean, which is IID over the training points, i.e.,

$$\tilde{r}_{m,i} = f(\tilde{\gamma}_{m,i}) + \epsilon, \quad \text{with} \quad \epsilon \sim \mathcal{N}\{0, \sigma_\epsilon^2\}. \quad (6)$$

The function f is unknown and has to be estimated via regression in the following. To define the underlying GP, we choose the squared exponential covariance function [22]

$$k(\tilde{\gamma}_{m,i}, \tilde{\gamma}_{m,j}) = \sigma_r^2 \exp\left(-\frac{1}{2\alpha^2}(\tilde{\gamma}_{m,i} - \tilde{\gamma}_{m,j})^2\right), \quad (7)$$

where $\tilde{\gamma}_{m,i}, \tilde{\gamma}_{m,j}$ denote averaged diffuseness values of the training or localization phase. The range variance σ_r^2, the noise variance σ_ϵ^2, and the length scale of the covariance function α are user-defined parameters. We construct the correlation matrix of the training points

$$\mathbf{K}(\tilde{\boldsymbol{\gamma}}_m) = [k(\tilde{\gamma}_{m,i}, \tilde{\gamma}_{m,j})]_{i,j} \quad \text{with} \quad 1 \le i, j \le N_{\text{train}} \quad (8)$$

and the correlation vector of the new estimate $\widehat{\gamma}_m$ obtained at node m with the training data

$$\mathbf{k}(\widehat{\gamma}_m, \tilde{\boldsymbol{\gamma}}_m) = \mathbf{k}(\tilde{\boldsymbol{\gamma}}_m, \widehat{\gamma}_m)^{\mathrm{T}} = [k(\widehat{\gamma}_m, \tilde{\gamma}_{m,i})]_i \quad (9)$$

with $1 \le i \le N_{\text{train}}$ from the correlation function (7). Now, we want to estimate the range r_m for a new value $\widehat{\gamma}_m$ calculated at the current sensor node m. We model the range of the training data and the range of a test estimate to be jointly normally distributed

$$\begin{bmatrix} \tilde{\mathbf{r}}_m \\ r_m \end{bmatrix} \sim \mathcal{N}\left\{\mathbf{0}, \begin{bmatrix} \mathbf{K}(\tilde{\boldsymbol{\gamma}}_m) + \sigma_\epsilon^2 \mathbf{I} & \mathbf{k}(\tilde{\boldsymbol{\gamma}}_m, \widehat{\gamma}_m) \\ \mathbf{k}(\widehat{\gamma}_m, \tilde{\boldsymbol{\gamma}}_m) & k(\widehat{\gamma}_m, \widehat{\gamma}_m) \end{bmatrix}\right\}. \quad (10)$$

This defines a GP, which is completely specified by its mean function (here identical to zero) and its covariance function. The predicted mean function can be computed as [22]

$$\hat{r}_m = \mathbf{k}(\widehat{\gamma}_m, \tilde{\boldsymbol{\gamma}}_m) \left(\mathbf{K}(\tilde{\boldsymbol{\gamma}}_m) + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \tilde{\mathbf{r}}_m \quad (11)$$

and the predictive variance by [22]

$$\begin{aligned} \mathbb{V}(\hat{r}_m) &= k(\widehat{\gamma}_m, \widehat{\gamma}_m) \dots \\ &\dots - \mathbf{k}(\widehat{\gamma}_m, \tilde{\boldsymbol{\gamma}}_m) \left(\mathbf{K}(\tilde{\boldsymbol{\gamma}}_m) + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \mathbf{k}(\tilde{\boldsymbol{\gamma}}_m, \widehat{\gamma}_m). \quad (12) \end{aligned}$$

Note that the correlation matrix $\mathbf{K}$ and the correlation vector $\mathbf{k}$ can be easily updated if new training data are available by appending new values of the covariance function. However, the

ISBN 978-90-827970-1-5 © EURASIP 2018

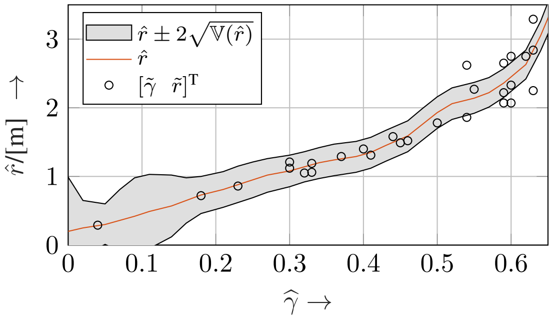

Fig. 1. Example of a resulting predicted mean function $\hat{r}$ (red); a hose illustrating $\mathbb{V}(r)$ (gray) and black circles corresponding to the training positions with the parameter settings described in the experimental part.

matrix inversion in (11) and (12) has to be calculated again. To circumvent this, the predicted mean (11) and variance (12) can be updated by online learning [23], [24], which is out of scope here. An exemplary regression curve is shown in Fig. 1. Note that only a small number of labeled training data points is needed for the algorithm to work, which, e.g., in a smart home environment, can be easily collected.

III. Localization

In the following section, the fusion of the obtained range estimates (see Sec. II-B) is developed for 2D localization. Note, that a similar procedure has been proposed by [10] for a non-weighted Least Squares (LS) problem. We define the position of the reference point $\mathbf{p}_m$ corresponding to sensor node m and the source position $\mathbf{q}$ as

$$\mathbf{p}_m = [p_{x,m}, p_{y,m}]^{\mathrm{T}} \quad \text{and} \quad \mathbf{q} = [q_x, q_y]^{\mathrm{T}}. \tag{13}$$

A. Weighted Least Squares Problem

The estimated range $\hat{r}_m$ of node m can be described by a circle around the node's reference point defined by the equation

$$\hat{r}_m^2 = r_m^2 + v_m = (q_x - p_{x,m})^2 + (q_y - p_{y,m})^2 + v_m, \tag{14}$$

with the true range r_m and additional zero-mean IID Gaussian measurement noise $v_m \sim \mathcal{N}\{0, \sigma_m^2\}$. The observation noise variance is dependent on the predictive variance of the regression step $\sigma_m^2 = g(\mathbb{V}(\hat{r}_m), \widehat{\gamma}_m)$, where $g(\cdot)$ is a user-defined weighting function of $\mathbb{V}(\hat{r}_m)$ depending on $\widehat{\gamma}_m$. For a compact notation, we stack the estimated source–node distances $\hat{r}_m$ in the vector $\hat{\mathbf{r}} \in \mathbb{R}_+^M$ and introduce the following substitutions

$$r_m^2 - \hat{r}_m^2 = R + c_{x,m} q_x + c_{y,m} q_y - c_{0,m}, \tag{15}$$

where we defined

$$R = q_x^2 + q_y^2, \quad c_{x,m} = -2p_{x,m}, \quad c_{y,m} = -2p_{y,m}, \tag{16}$$

$$c_{0,m} = \hat{r}_m^2 - p_{x,m}^2 - p_{y,m}^2. \tag{17}$$

If M nodes contribute to an estimate, (14) can be equivalently represented in matrix notation by the likelihood as

$$p\left(\hat{\mathbf{r}} | \boldsymbol{\theta}(\mathbf{q})\right) = \frac{1}{(2\pi)^M \sqrt{\det \boldsymbol{\Sigma}_M}} \cdots \tag{18}$$

$$\cdots \exp\left(-\frac{1}{2}(\mathbf{c}_{0,M} - \mathbf{C}_M \boldsymbol{\theta})^{\mathrm{T}} \boldsymbol{\Sigma}_M^{-1} (\mathbf{c}_{0,M} - \mathbf{C}_M \boldsymbol{\theta})\right).$$

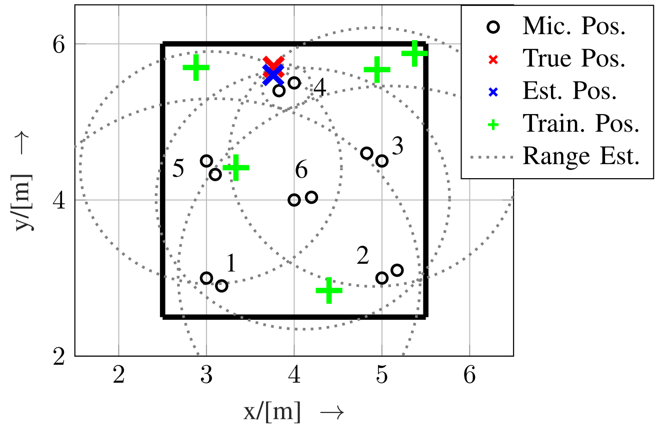

Fig. 2. Example for a localization result within the region of interest, marked by the black box.

Here, the following notations have been applied

$$\mathbf{C}_M = \begin{bmatrix} 1 & c_{x,1} & c_{y,1} \\ & \cdots & \\ 1 & c_{x,M} & c_{y,M} \end{bmatrix}, \quad \mathbf{c}_{0,M} = \begin{bmatrix} c_{0,1} \\ \vdots \\ c_{0,M} \end{bmatrix}, \quad \boldsymbol{\theta} = \begin{bmatrix} R \\ q_x \\ q_y \end{bmatrix},$$

and

$$\boldsymbol{\Sigma}_M = \operatorname{diag}\left\{\left[\sigma_1^2, \ldots, \sigma_M^2\right]\right\}.$$

This yields the maximum likelihood optimization problem

$$\hat{\boldsymbol{\theta}}_M = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\; p\left(\hat{\mathbf{r}} | \boldsymbol{\theta}(\mathbf{q})\right). \tag{19}$$

Its solution is the WLS estimate [25]

$$\hat{\boldsymbol{\theta}}_M = \mathbf{S}[M] \mathbf{C}_M^{\mathrm{T}} \boldsymbol{\Sigma}_M^{-1} \mathbf{c}_{0,M}, \tag{20}$$

which yields an estimate of the source position $\hat{\mathbf{q}}_M = [\hat{\theta}_{2,M}, \hat{\theta}_{3,M}]^{\mathrm{T}} = [\hat{q}_x, \hat{q}_y]^{\mathrm{T}}$ as the non-redundant part of $\hat{\boldsymbol{\theta}}_M$. Here we used the abbreviation $\mathbf{S}[M] = \left(\mathbf{C}_M^{\mathrm{T}} \boldsymbol{\Sigma}_M^{-1} \mathbf{C}_M\right)^{-1}$.

An exemplary localization result is shown in Fig. 2.

B. Sequential Least Squares

In ASNs, the computational power of the sensor nodes is usually low and data rates for communication are limited. Therefore, we use a sequential realization of the WLS estimator to distribute the computational load over the network. The sequential WLS estimator has a lower computational complexity than the corresponding batch estimate, because no matrix inversion is necessary [25]. At the same time, the amount of data which has to be transmitted over the network is very low.

Initial values for $\mathbf{S}[m-1]$ and $\hat{\boldsymbol{\theta}}[m-1]$ have to be specified to start the sequential node updating. This is done by collecting a total of m' estimates and conduct a batch estimation step (20) using the m' estimates. The initial estimator covariance $\mathbf{S}[m']$ is given by

$$\mathbf{S}[m-1] = \mathbf{S}[m'] = \left(\mathbf{C}_{m'}^{\mathrm{T}} \boldsymbol{\Sigma}_{m'}^{-1} \mathbf{C}_{m'}\right)^{-1} \tag{21}$$

and the initial estimate $\hat{\boldsymbol{\theta}}[m']$ by

$$\hat{\boldsymbol{\theta}}[m-1] = \hat{\boldsymbol{\theta}}[m'] = \mathbf{S}[m'] \mathbf{C}_{m'}^{\mathrm{T}} \boldsymbol{\Sigma}_{m'}^{-1} \mathbf{c}_{0,m'}. \tag{22}$$

ISBN 978-90-827970-1-5 © EURASIP 2018

The vector containing the position of node m is denoted as

$$\mathbf{c}[m] = [1, c_{x,m}, c_{y,m}]^{\mathrm{T}}. \tag{23}$$

Node m obtains the covariance matrix estimate $\mathbf{S}[m-1]$ and the estimated parameter vector $\hat{\boldsymbol{\theta}}[m-1]$ of the LS estimator from the previous node $m-1$. The gain factor [25]

$$\mathbf{G}[m] = \frac{\mathbf{S}[m-1]\mathbf{c}[m]}{\sigma_m^2 + \mathbf{c}^{\mathrm{T}}[m]\mathbf{S}[m-1]\mathbf{c}[m]} \tag{24}$$

is needed for the update of the WLS estimate [25]

$$\hat{\boldsymbol{\theta}}[m] = \hat{\boldsymbol{\theta}}[m-1] + \mathbf{G}[m]\left(c_{0,m} - \mathbf{c}[m]^{\mathrm{T}}\hat{\boldsymbol{\theta}}[m-1]\right) \tag{25}$$

and for the update of the estimator covariance [25]

$$\mathbf{S}[m] = \left(\mathbf{I} - \mathbf{G}[m]\mathbf{c}^{\mathrm{T}}[m]\right)\mathbf{S}[m-1]. \tag{26}$$

The source position estimate $\hat{\mathbf{q}}_m = \left[\hat{\theta}_2[m], \hat{\theta}_3[m]\right]^{\mathrm{T}}$ at node m is finally obtained from the non-redundant part of $\hat{\boldsymbol{\theta}}[m]$.

The data which have to be transmitted in the localization phase to a successive node are $2m$ real numbers (m radii and m variances, as $\boldsymbol{\Sigma}_m$ is diagonal) if $m < m'$ and 8 real numbers for $m \geq m'$ (the first element of $\hat{\boldsymbol{\theta}} \in \mathbb{R}^3$ is redundant and $\mathbf{S} \in \mathbb{R}^{3\times 3}$ is symmetric). The localization after completing the training phase is summarized in Algorithm 1.

IV. Simulation Study

A. Scenario

We investigated the performance of the algorithm within an enclosure simulated by the image-source method [26] using the RIR generator [27]. The simulated enclosure was of dimensions $10\,\mathrm{m} \times 8\,\mathrm{m} \times 10\,\mathrm{m}$, from which we chose a 2D $3\,\mathrm{m} \times 3.5\,\mathrm{m}$ region of interest in the x-y-plane (see Fig. 2) with all sources and microphones being placed at a height of $z = 2\,\mathrm{m}$. A total of $M = 6$ nodes, each containing a microphone pair with spacing $d_{\text{mic}} = 0.2\,\mathrm{m}$ have been randomly distributed over the region of interest. The reference point of a sensor node is the center of the microphone pair of the node. For each test position, a speech signal of duration $5\,\mathrm{s}$ was convolved with the simulated RIRs and transformed into the STFT domain at each node, by using a von Hann window of length $25\,\mathrm{ms}$ and frame shift of $10\,\mathrm{ms}$. Using the frequency interval $[f_{\min}, f_{\max}] = [125\,\mathrm{Hz}, 3500\,\mathrm{Hz}]$, the averaged diffuseness $\widehat{\gamma}$ was computed for each node, with a strong smoothing ($\lambda = 0.95$) to alleviate the influence of speech pauses. These parameters hold for the training as well as for the localization phase of the algorithm. We chose the following parameters for the regression described by (7) and (10), $\alpha = 0.1$, $\sigma_r^2 = 1.5$, and $\sigma_\epsilon^2 = 0.01$.

For large values of $\widehat{\gamma}$, the slope of the regression function (11) becomes very steep (see Fig. 1) yielding range estimates, which are very sensitive to small variations of $\widehat{\gamma}$. We account for that by penalizing these estimates with a small weight, corresponding to a high variance. This is represented by

$$\sigma_m^2 = g(\mathbb{V}(\hat{r}_m), \widehat{\gamma}_m) = \begin{cases} \mathbb{V}(\hat{r}_m), & \text{if } \widehat{\gamma}_m \leq 0.65 \\ 10, & \text{else} \end{cases}. \tag{27}$$

The initialization (21) and (22) of the sequential LS estimator was done by the data collected from $m' = 3$ sensor nodes.

Algorithm 1 Localization for node m

```
Input: $\mathbf{K}(\tilde{\gamma})$, $\mathbf{k}(\widehat{\gamma}_m, \tilde{\gamma})$
  Compute range estimate $\hat{r}_m$ (11) and variance $\sigma_m^2$ (12), (27)
  if $m < m'$ then
    Send range estimate $[\hat{r}_1, \dots, \hat{r}_m]^{\mathrm{T}}$ and $\boldsymbol{\Sigma}_m$ to next node
  else if $m = m'$ then
    Initialize with (21) and (22)
    Send estimates $\mathbf{S}[m']$, $\hat{\boldsymbol{\theta}}[m']$ to next node
  else
    Receive $\mathbf{S}[m-1]$ and $\hat{\boldsymbol{\theta}}[m-1]$
    Update weighted LS estimate (24), (25) and (26)
    Send estimates $\mathbf{S}[m]$, $\hat{\boldsymbol{\theta}}[m]$ to next node
  end if
Output: $\hat{\mathbf{q}}_m$, if $m \geq m'$
```

B. Results

We drew $N_{\text{runs}} = 10$ times N_{train} random positions for the training of the algorithm and evaluated the trained algorithm at each run with $N_{\text{eval}} = 100$ test positions. The localization error was evaluated by computing the Euclidean distance between the estimated and the true source position for each run and evaluation position, with the average being computed as

$$e = \frac{1}{N_{\text{runs}} N_{\text{eval}}} \sum_{i=1}^{N_{\text{runs}}} \sum_{j=1}^{N_{\text{eval}}} \|\hat{\mathbf{q}}_{i,j} - \mathbf{q}_{i,j}\|_2. \tag{28}$$

We repeated the experiment with $N_{\text{train}} = 10$ training positions for different reverberation times $T_{60} = 0.3\,\mathrm{s}, 0.5\,\mathrm{s}, 0.7\,\mathrm{s}, 1\,\mathrm{s}$. The results are depicted in Fig. 3 as a function of the number m of nodes used for the sequential update of the WLS estimator. Note that the sequential estimator is not an approximation of the batch version but an exact reformulation of it. Therefore, the results of the sequential update, as illustrated in the figure, are the same as for a batch algorithm based on the same number of nodes. By inspection of the resulting averaged localization errors e, we can conclude that the proposed algorithm works accurately for a wide range of reverberant environments. Furthermore, we investigated the effect of the number of training data points. To this end, we conducted experiments as described above, with $T_{60} = 0.5\,\mathrm{s}$ and $N_{\text{train}} = 1, 2, 5, 10, 20$. The results depicted in Fig. 4 get better for increasing N_{train}. However, this effect gets smaller and more or less vanishes at $N_{\text{train}} = 10$. Finally we investigated the influence of additive noise on the localization performance. To this end, we carried out experiments with an SNR at the microphones varying between $-30\,\mathrm{dB}$ and $30\,\mathrm{dB}$ with $N_{\text{train}} = 10$ and $T_{60} = 0.5\,\mathrm{s}$. The results using all six nodes are depicted in Fig. 5. It can be seen that the algorithm performs well for a broad range of SNR values and that the performance starts to degrade significantly at an SNR of about $-10\,\mathrm{dB}$.

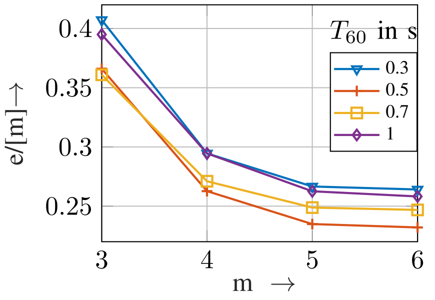

Fig. 3. Localization error e of the sequential LS estimator for varying T_{60}.

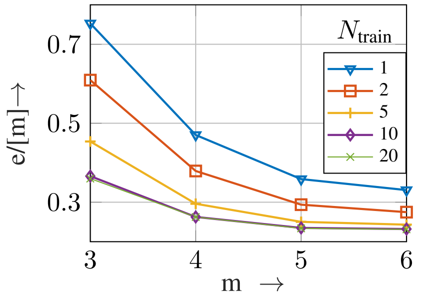

Fig. 4. Localization error e of the sequential LS estimator for varying N_{train}.

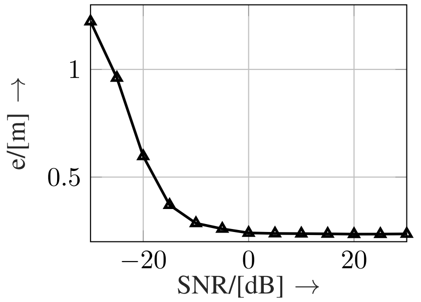

Fig. 5. Localization error e of the final sequential LS estimate for varying SNR.

V. Conclusions

We proposed a localization method for ASNs, which works in a wide range of acoustic scenarios including very reverberant and noisy ones. The algorithm performs well with only a small number of training data points and is of low computational complexity. The data rate for the communication between the nodes is very low, which allows for a distributed implementation. The efficacy of the approach has been shown in an enclosure simulated by the image-source method. Future work will include experiments with signals measured in a real acoustic environment, distributed training and the investigation of scenarios with multiple sources.

References

[1] A. Bertrand, “Applications and trends in wireless acoustic sensor networks: A signal processing perspective,” in *Proc. of the IEEE Symp. on Commun. and Vehicular Technology (SCVT)*, Ghent, Belgium, Nov. 2011, pp. 1–6.

[2] G. Han, H. Xu, T. Q. Duong, J. Jiang, and T. Hara, “Localization algorithms of Wireless Sensor Networks: a survey,” *Telecommun. Systems*, vol. 52, no. 4, pp. 2419–2436, Apr. 2013.

[3] A. Alexandridis and A. Mouchtaris, “Multiple sound source location estimation and counting in a wireless acoustic sensor network,” in *IEEE Workshop on Applicat. of Signal Process. to Audio and Acoust. (WASPAA)*, New Paltz, NY, USA, Oct. 2015, pp. 1–5.

[4] O. Schwartz and S. Gannot, “Speaker Tracking Using Recursive EM Algorithms,” *IEEE/ACM Trans. on Audio, Speech, and Language Process.*, vol. 22, no. 2, pp. 392–402, Feb. 2014.

[5] Y. Dorfan and S. Gannot, “Tree-Based Recursive Expectation-Maximization Algorithm for Localization of Acoustic Sources,” *IEEE/ACM Trans. on Audio, Speech, and Language Process.*, vol. 23, no. 10, pp. 1692–1703, Oct. 2015.

[6] J. H. DiBiase, “A High-Accuracy, Low-Latency Technique for Talker Localization in Reverberant Environments Using Microphone Arrays,” PHD Thesis, Brown University, Providence, Rhode Island, May 2000.

[7] A. Griffin, A. Alexandridis, D. Pavlidi, Y. Mastorakis, and A. Mouchtaris, “Localizing multiple audio sources in a wireless acoustic sensor network,” *Signal Process.*, vol. 107, pp. 54–67, Feb. 2015.

[8] C. Meesookho, U. Mitra, and S. Narayanan, “On Energy-Based Acoustic Source Localization for Sensor Networks,” *IEEE Trans. on Signal Processing*, vol. 56, no. 1, pp. 365–377, Jan. 2008.

[9] Y. Huang, J. Benesty, G. Elko, and R. Mersereati, “Real-time passive source localization: a practical linear-correction least-squares approach,” *IEEE Trans. on Speech and Audio Process.*, vol. 9, no. 8, pp. 943–956, Nov. 2001.

[10] C. Meesookho and S. Narayanan, “Distributed Range Difference Based Target Localization in Sensor Network,” in *Conf. Rec. of the Thirty-Ninth Asilomar Conf. on Signals, Systems and Comput.*, Pacific Grove, CA, USA, 2005, pp. 205–209.

[11] E. Larsen, C. Schmitz, C. Lansing, W. O’Brien, B. Wheeler, and A. Feng, “Acoustic scene analysis using estimated impulse responses,” in *Conf. Rec. of the Thirty-Seventh Asilomar Conf. on Signals, Systems and Comput.*, Pacific Grove, CA, USA, Nov. 2003, pp. 725–729.

[12] F. Keyrouz, “Binaural range estimation using Head Related Transfer Functions,” in *IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, San Diego, CA, USA, Sep. 2015, pp. 89–94.

[13] Y. Hioka, K. Niwa, S. Sakauchi, K. Furuya, and Y. Haneda, “Estimating Direct-to-Reverberant Energy Ratio Using D/R Spatial Correlation Matrix Model,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 19, no. 8, pp. 2374–2384, Nov. 2011.

[14] S. Vesa, “Binaural Sound Source Distance Learning in Rooms,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 17, no. 8, pp. 1498–1507, Nov. 2009.

[15] Y.-C. Lu and M. Cooke, “Binaural Estimation of Sound Source Distance via the Direct-to-Reverberant Energy Ratio for Static and Moving Sources,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 18, no. 7, pp. 1793–1805, Sep. 2010.

[16] E. Georganti, T. May, S. van de Par, and J. Mourjopoulos, “Sound Source Distance Estimation in Rooms based on Statistical Properties of Binaural Signals,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 21, no. 8, pp. 1727–1741, Aug. 2013.

[17] E. Georganti, T. May, S. van de Par, A. Harma, and J. Mourjopoulos, “Speaker Distance Detection Using a Single Microphone,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 19, no. 7, pp. 1949–1961, Sep. 2011.

[18] I. McCowan, M. Lincoln, and I. Himawan, “Microphone Array Shape Calibration in Diffuse Noise Fields,” *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 16, no. 3, pp. 666–670, Mar. 2008.

[19] A. Schwarz and W. Kellermann, “Coherent-to-Diffuse Power Ratio Estimation for Dereverberation,” *IEEE/ACM Trans. on Audio, Speech, and Language Process.*, vol. 23, no. 6, pp. 1006–1018, Jun. 2015.

[20] N. Cressie, “The origins of Kriging,” *Math. Geology*, vol. 22, no. 3, pp. 239–252, Apr. 1990.

[21] M. Umer, L. Kulik, and E. Tanin, “Kriging for Localized Spatial Interpolation in Sensor Networks,” in *Scientific and Statistical Database Management*, B. Ludscher and N. Mamoulis, Eds. Springer Berlin Heidelberg, 2008, vol. 5069, pp. 525–532.

[22] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*, ser. Adaptive computation and machine learning. Cambridge: MIT Press, 2006.

[23] L. Csat and M. Opper, “Sparse representation for Gaussian process models,” *Advances in Neural Inform. Process. Systems*, vol. 13, pp. 444–450, 1994.

[24] F. Perez-Cruz, S. Van Vaerenbergh, J. J. Murillo-Fuentes, M. Lazaro-Gredilla, and I. Santamaria, “Gaussian Processes for Nonlinear Signal Processing: An Overview of Recent Advances,” *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 40–50, Jul. 2013.

[25] S. M. Kay, *Fundamentals of statistical signal processing - Estimation Theory*, ser. Prentice Hall signal processing series. Englewood Cliffs, N.J: Prentice-Hall PTR, 1993, vol. 1.

[26] J. B. Allen and D. A. Berkley, “Image method for efficiently simulating small-room acoustics,” *J. Acoust. Soc. of America*, vol. 65, no. 4, pp. 943–950, Apr. 1979.

[27] E. A. P. Habets, “Room Impulse Response Generator,” Int. Audio Laboratories, Tech. Rep., Sep. 2010.