# Loudspeaker and Listening Position Estimation using Smart Speakers

Nielsen, Jesper Kjær

*Published in:*
2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

*DOI (link to publication from Publisher):*
10.1109/ICASSP.2018.8462302



*Publication date:*
2018

*Document Version*
Accepted author manuscript, peer reviewed version



*Citation for published version (APA):*
Nielsen, J. K. (2018). Loudspeaker and Listening Position Estimation using Smart Speakers. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 81-85). Article 8462302 IEEE Press. https://doi.org/10.1109/ICASSP.2018.8462302

# LOUDSPEAKER AND LISTENING POSITION ESTIMATION USING SMART SPEAKERS

*Jesper Kjær Nielsen*

Audio Analysis Lab, CREATE, Aalborg University, Denmark
Acoustic Research, Bang & Olufsen A/S, Denmark
jkn@create.aau.dk

## ABSTRACT

Recently, so-called smart speakers have been introduced and they include a microphone array. One potential application of such a smart speaker is to use it for calibrating a larger audio system which the speaker is a part of. In this paper, we propose a method to perform this calibration using one or several smart speakers. Specifically, a map is estimated of the sensors and sound sources. As opposed to existing methods, the proposed method can create this map for both synchronised and unsynchronised sound sources by taking the different localisation errors into account. We show that this gives more accurate estimates than assuming identical estimation errors, and that existing methods are outperformed in terms of estimation accuracy for various noise levels and reverberation times.

***Index Terms—*** Array processing, Procrustes analysis, source and sensor calibration

## 1. INTRODUCTION

The listening experience is highly influenced by the position of the loudspeakers relative to the listener. For example, the two loudspeakers and the listener should ideally be placed on the vertices of an even sided triangle in a stereo setup, and the loudspeakers should be placed at certain angles on a circle centred on the listening position in a surround sound setup [1]. Unfortunately, the loudspeaker and listening positions are often not at their ideal position since other interior design considerations may take higher priority. Moreover, listeners are seldom willing to move the loudspeakers if they temporarily want to move the optimal listening position (the so-called sweet spot) from one point to another. However, if the positions of the loudspeakers and the listener are known, signal processing algorithms can to a certain extent compensate for the non-ideal positions and move the sweep spot. The traditional way of calibrating an audio system to one or several listening positions is to place a microphone at the listening position(s) and then run a calibration sequence. This procedure allows the audio system to compensate for the distances from the loudspeakers to the listening position(s) and for some aspects of the room, but does not produce a map over the loudspeakers. The latter is required for rendering object based audio such as specified by the MPEG-H standard [2]. Also, the calibration is often only performed as a part of the initial setup of the system since it requires some effort by the listener or a trained installer.

Recently, loudspeakers such as the Amazon Echo, the Google Home, and the Apple HomePod come equipped with built-in microphones. This allows the loudspeakers to be used for many other applications than just standard audio playback, and they are, therefore, often referred to as *smart speakers*. One potential application of smart speakers is the calibration of a larger audio system which the smart speaker is a part of. By using the microphones within the smart speaker, other loudspeakers and the listener can be located and placed in a map. If multiple smart speakers are connected to the same audio system, they all produce local maps which can be combined into a global map.

Acoustic source and sensor geometry calibration has been a research topic for several decades. A lot of work has focused on creating a map for individual sensors and sources which were not necessarily synchronised (see, e.g., [3–8]). However, at least four sources, four sensors, and a total of at least ten transducers (sources + sensors) are required to solve the geometry calibration in the synchronised case [4], and many current audio reproduction systems consist of fewer sources and sensors than that. In order to go below this limit, prior information must be included in the problem. Such prior information can be in the form of the structure of some of the sources and sensors. If smart speakers are a part of the acoustic network, the sensors and sources are organised in subarrays where the local geometry is known. If the knowledge of the local geometry is taken into account, more accurate estimates can be obtained with only a few subarrays. Exactly this was recently demonstrated in [7, 8], but the proposed multidimentional unfolding (MDU) method requires many sources and sensors to work. When this is the case, however, MDU outperforms existing methods.

The self-calibration problem using subarrays is typically referred to as interarray calibration [3] or array configuration calibration [9], and a number of methods have been proposed under various assumptions. A recent approach in [10] (and later improved in [11]) produces a high estimation accuracy, but requires that the raw microphone data (or a sparse spike representation thereof) are exchanged between the subarrays. Moreover, the method only assumes unsynchronised sources and does not take into account that the various subarrays cannot estimate the source positions with the same accuracy. As demonstrated in [12], a better estimation accuracy and robustness to outliers can be obtained if these uncertainties are taken into account. Whereas [11, 12] assume only unsynchronised sources at unknown locations, [13] assumes that each subarray has exactly one synchronised source whose location is known relative to the subarray. This corresponds to a scenario where an audio system consists of only smart speakers.

In this paper, we propose a method for creating a map over synchronised sources (e.g., loudspeakers), unsynchronised sources (e.g., listener(s)), and sensor subarrays. The method does not require that the raw microphone data are transmitted between the subarrays or to a central processing unit, and it also takes localisation errors into account when combining the estimated maps of each subarray into a global map. As opposed to existing methods, the proposed method works for a combination of synchronised and unsynchronised sources, and the relative positions of the synchronised sources do not have to be known. Finally, the method does not require at least five sources as in [12], but works even for a simple stereo setup.

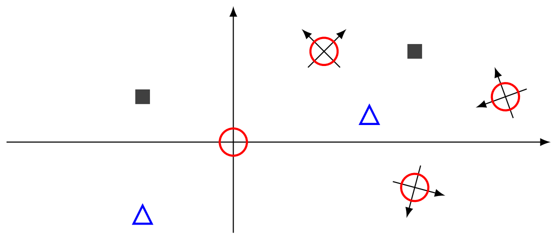

**Fig. 1**. Illustration of the general setup. The circles are sensor arrays with their own local coordinate system, the filled squares are the synchronised sources, and the open triangles are sources emitting an unknown or unsynchronised source signal.

## 2. THE PROBLEM

Fig. 1 illustrates the general problem considered in this paper. In the figure, the circles are sensor arrays (e.g., microphone arrays), the filled squares are the synchronised sources (e.g., loudspeakers), and the open triangles are sources emitting an unknown or unsynchronised source signal (e.g., a talking person or a mobile phone). The sensor arrays and the sources may or may not be co-located, and the orientation of the sensor arrays are unknown. Without loss of generality, we have also assumed that the reference coordinate system is the coordinate system of one of the sensor arrays. Additionally, we have the following restrictions for the problem.

1. The sensor arrays and the synchronized sources are synchronised to within a few tens of microseconds to a master (e.g., a tv). In a loudspeaker system where smart speakers are used, this is not an unrealistic assumption since synchronisation is required to reproduce spatial audio faithfully.
2. Due to a limited data channel, the raw sensor data cannot be sent directly to the master. Local processing is, therefore, necessary in the sensor arrays.

Under these restrictions, the problem considered in this paper is to reconstruct the map from the sensor recordings. We propose solving this problem using the following two-step algorithm.

1. In turn, the sources emit a calibration signal while the $M$ sensor arrays estimate the positions of these sources in their own local coordinate systems. In addition to the position estimates, the sensor arrays also compute quality matrices describing the accuracy of the estimated positions.
2. The position estimates of the sources in the $M$ local coordinate systems are transmitted to the master along with the quality matrices. The master then rotates and translates the local coordinate systems so that they fit as well as possible. The quality matrices are used in this process to ensure that the most accurate estimates have a bigger contribution than the less-accurate estimates.

In the next two sections, we go into details with these steps. Due to space constraints and for the sake of clarity, we only describe the 2D-version of the algorithm, but the principles described here can also be applied to the 3D-case.

## 3. SOURCE LOCALISATION

Many source localisation algorithms already exist in the scientific literature for various array geometries. In principle, any array geometry can be used as long as at least three[1] sensors (not on the same line) are used. Let an array have $K \geq 3$ sensors, each with their own direction-dependent and known impulse response vector $\boldsymbol{h}_k(\theta) \in \mathbb{R}^{M_k}$ where $\theta$ is the DOA of a source. The source emits a signal which is received by each sensor $\eta_k$ samples later. In source localisation, the source signal is often (implicitly) assumed to be $N$-periodic [14] so that a time-shift of an $N$-length signal corresponds to a phase-shift in the frequency domain. Note that this assumption is easily satisfied for the synchronised sources since we can design the calibration signal. For a time-shift $\eta_k$, any $N$-periodic signal can be written as [14]

$$s(n - \eta_k) = \sum_{l=-L}^{L} \alpha_l \exp(jl\omega_0(n - \eta_k)) \quad (1)$$

with $\alpha_l = \alpha^*_{-l}$ being a complex amplitude ($\alpha_0$ is real), $\omega_0 = 2\pi/N$ is the fundamental frequency, and $L = \lfloor N/2 \rfloor$ is the maximum number of harmonic components. To facilitate a fast implementation, the shifted source signal for $n = 0, 1, \ldots, N-1$ can also be rewritten in terms of the DFT matrix $\boldsymbol{F} = \{\exp(j2\pi nr/N)\}_{n,r=0,\ldots,N-1}$ as

$$\boldsymbol{s}(\eta_k) = N^{-1}\boldsymbol{F}\boldsymbol{Q}(\eta_k)\boldsymbol{F}^H\boldsymbol{s}(0) \quad (2)$$

where $\boldsymbol{Q}(\eta_k) = \mathrm{diag}(\boldsymbol{q}(\eta_k))$. The definition of $\boldsymbol{q}(\eta_k)$ depends on whether $N$ is even or not. If $N$ is even, then

$$\begin{aligned}\boldsymbol{q}(\eta_k) = \big[1 \quad &\exp(-j\omega_0\eta_k) \quad \cdots \quad \exp(-j(L-1)\omega_0\eta_k) \\ \cos(L\omega_0\eta_k) \quad &\exp(j(L-1)\omega_0\eta_k) \quad \cdots \quad \exp(j\omega_0\eta_k)\big]^T . \quad (3)\end{aligned}$$

Conversely, if $N$ is uneven, then

$$\begin{aligned}\boldsymbol{q}(\eta_k) = \big[1 \quad &\exp(-j\omega_0\eta_k) \quad \cdots \quad \exp(-jL\omega_0\eta_k) \\ &\exp(jL\omega_0\eta_k) \quad \cdots \quad \exp(j\omega_0\eta_k)\big]^T . \quad (4)\end{aligned}$$

Each sensor records $N$ samples which are a noisy version of the shifted source signal convolved with the corresponding sensor response. This can be written as

$$\boldsymbol{y}_k = \frac{\beta}{\eta_k}\boldsymbol{H}_k(\theta)\boldsymbol{s}(\eta_k) + \boldsymbol{e}_k \quad (5)$$

where $\beta > 0$ is an unknown gain and $\boldsymbol{H}_k(\theta)$ is a convolution matrix. Since the source signal is $N$-periodic, the convolution matrix is circulant and is, therefore, diagonalised by the DFT matrix $\boldsymbol{F}$. Thus, we have that

$$\begin{aligned}\boldsymbol{y}_k &= \frac{\beta}{\eta_k}\frac{1}{N}\boldsymbol{F}\boldsymbol{\Lambda}_k(\theta)\boldsymbol{F}^H\frac{1}{N}\boldsymbol{F}\boldsymbol{Q}(\eta_k)\boldsymbol{F}^H\boldsymbol{s}(0) + \boldsymbol{e}_k \quad (6)\\ &= \boldsymbol{G}_k(\boldsymbol{p})\boldsymbol{s}(0)\beta + \boldsymbol{e}_k \quad (7)\end{aligned}$$

where $\boldsymbol{\Lambda}_k(\theta)$ is a diagonal matrix containing the DFT of $\boldsymbol{h}_k(\theta)$, $\boldsymbol{G}_k(\boldsymbol{p}) = \frac{1}{N\eta_k}\boldsymbol{F}\boldsymbol{\Lambda}_k(\theta)\boldsymbol{Q}(\eta_k)\boldsymbol{F}^H$, and $\boldsymbol{p}$ is the position of the source. To estimate this source position, we seek the parameters which minimise the squared error $\sum_{k=1}^{K}\boldsymbol{e}_k^T\boldsymbol{e}_k$. Equivalently, but more efficiently, the minimisation can be performed by minimising the residual sum of squares w.r.t. the source position $\boldsymbol{p}$. Thus, we first replace the linear parameters in (7) with their least-squares estimates and then minimise the squared residual. When the source signal $\boldsymbol{s}(0)$ is known, $\beta$ is the linear parameter. Conversely, we cannot distinguish between $\boldsymbol{s}(0)$ and $\beta$ when both are unknown so the product $\boldsymbol{s}(0)\beta$ is the linear parameters in the case of an unknown source signal.

The described signal model and estimation procedure can be used for any array geometry. In the experiments, we have used a uniform circular array (UCA) since the DOA estimation performance is independent of the direction of the source [15, 16] and fast estimation algorithms for it exist [17]. Moreover, a UCA is often used in smart speakers.

[1] In 3D, at last four sensors (not in the same plane) are required.

### 3.1. Quality matrices

The quality matrices represent how accurately the sources are estimated by the sensor array. This information is very useful when the local coordinate systems are combined into a global coordinate system. It is also absolutely essential when estimates of synchronised and unsynchronised sources are mixed since we can estimate the range of the former much more accurately than for the latter. As we detail below, we compute the quality matrices from the observed Fisher information matrices (FIMs). We focus the attention to the case of synchronised sources, but the same derivation can be followed for the case of unsynchronised sources.

We assume that the noise $\boldsymbol{e}_k$ is white and Gaussian, so that the recorded data are distributed as $\boldsymbol{y}_k \sim \mathcal{N}(\boldsymbol{\mu}_k(\boldsymbol{\vartheta}), \sigma^2 \boldsymbol{I}_N)$ where $\boldsymbol{\mu}_k(\boldsymbol{\vartheta}) = \boldsymbol{G}_k(\boldsymbol{p})\boldsymbol{s}(0)\beta$ and $\boldsymbol{\vartheta} = \begin{bmatrix}\beta & \boldsymbol{p}^T\end{bmatrix}^T$. The FIM is then defined as [18, Sec. 3.9]

$$\mathcal{I}(\boldsymbol{\vartheta}) = \frac{1}{\sigma^2}\sum_{k=1}^{K}\left(\frac{\partial \boldsymbol{\mu}_k(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}^T}\right)^T \frac{\partial \boldsymbol{\mu}_k(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}^T} = \frac{1}{\sigma^2}\begin{bmatrix} a & \boldsymbol{b}^T \\ \boldsymbol{b} & \boldsymbol{C}\end{bmatrix} . \quad (8)$$

The inverse FIM is, therefore, given by

$$\mathcal{I}^{-1}(\boldsymbol{\vartheta}) = \sigma^2 \begin{bmatrix} \times & \times \\ \times & \left(\boldsymbol{C} - \boldsymbol{b}a^{-1}\boldsymbol{b}^T\right)^{-1}\end{bmatrix} \quad (9)$$

from with we can extract the inverse quality matrix to

$$\boldsymbol{V}^{-1} = \sigma^{-1}\left(\boldsymbol{C} - \boldsymbol{b}a^{-1}\boldsymbol{b}^T\right)^{1/2} . \quad (10)$$

The observed FIM is obtained from the FIM by replacing the true parameter values with their estimates. Using the observed FIM as an estimate of the unknown FIM works in our experience well, unless the estimated source location is far from the true one. This is much more likely to happen for unsynchronised sources since the range estimate is very uncertain when the array radius is small relative to the range. A simple heuristic fix for this is to assume a big value for the range estimates so that effectively only the DOA estimates are used in fitting.

## 4. FITTING

So far, we have described how each sensor array computes estimates of the source positions and how the associated quality matrices are computed. In this section, we combine all this information into one global map of all the sensors and sources.

Assume that the true coordinates of $S$ sources in a reference coordinate system are given as the columns in the matrix $\boldsymbol{X} \in \mathbb{R}^{2\times S}$. In the coordinate system of the $m$'th sensor array, these global coordinates are observed rotated and translated as

$$\boldsymbol{X}_m = \boldsymbol{Q}_m \boldsymbol{X} + \boldsymbol{t}_m \boldsymbol{1}^T \quad (11)$$

where $\boldsymbol{Q}_m \in \mathbb{R}^{2\times 2}$ and $\boldsymbol{t}_m \in \mathbb{R}^{2\times 1}$ are a rotation matrix and a translation vector, respectively. Without loss of generality, we assume that the coordinate system of sensor array 1 is the reference coordinate system so that $\boldsymbol{Q}_1 = \boldsymbol{I}_2$ and $\boldsymbol{t}_1 = \boldsymbol{0}$. Unfortunately, we do not observe $\boldsymbol{X}_m$ directly, but only the noisy version

$$\boldsymbol{y}_m = \mathrm{vec}(\boldsymbol{Y}_m) = \mathrm{vec}(\boldsymbol{X}_m) + \boldsymbol{W}_m \boldsymbol{\epsilon}_m \quad (12)$$

where $\mathrm{vec}(\cdot)$ is the vectorisation operator, $\boldsymbol{W}_m \in \mathbb{R}^{2S\times 2S}$ is a block diagonal matrix of the form $\boldsymbol{W}_m = \mathrm{diag}\left(\boldsymbol{V}_{m1}, \dots, \boldsymbol{V}_{mS}\right)$, and $\boldsymbol{\epsilon}_m = \mathrm{vec}(\boldsymbol{E}_m) \in \mathbb{R}^{2S\times 1}$. The quality matrix $\boldsymbol{V}_{ms} \in \mathbb{R}^{2\times 2}$ is given by (10). Combining (11) and (12) gives the signal model

$$\boldsymbol{y}_m = \begin{cases} \boldsymbol{x} + \boldsymbol{W}_1 \boldsymbol{\epsilon}_1 & m = 1 \\ \begin{bmatrix}\boldsymbol{A}(\boldsymbol{Q}_m) & \boldsymbol{B}\end{bmatrix}\begin{bmatrix}\boldsymbol{x} \\ \boldsymbol{t}_m\end{bmatrix} + \boldsymbol{W}_m \boldsymbol{\epsilon}_m & m = 2, \dots, M \end{cases}$$

where $\boldsymbol{A}(\boldsymbol{Q}_m) = \boldsymbol{I}_S \otimes \boldsymbol{Q}_m$, $\boldsymbol{B} = \boldsymbol{1} \otimes \boldsymbol{I}_2$, $\boldsymbol{x} = \mathrm{vec}(\boldsymbol{X})$, and $\otimes$ denotes the Kronecker product. The task is now to estimate $\boldsymbol{X}$ given the quality matrices in $\{\boldsymbol{W}_m\}_{m=1}^M$ and the observations $\{\boldsymbol{Y}_m\}_{m=1}^M$. By stacking all the $\boldsymbol{y}_m$'s on top of each other for $m = 1, \dots, M$, we obtain the signal model

$$\boldsymbol{y} = \boldsymbol{H}(\boldsymbol{Q})\boldsymbol{z} + \boldsymbol{W}\boldsymbol{\epsilon} \quad (13)$$

where $\boldsymbol{Q} = \begin{bmatrix}\boldsymbol{Q}_2^T & \cdots & \boldsymbol{Q}_M^T\end{bmatrix}^T$ and

$$\boldsymbol{H}(\boldsymbol{Q}) = \begin{bmatrix}\boldsymbol{I}_{2S} & \boldsymbol{0} \\ \boldsymbol{G}(\boldsymbol{Q}) & \boldsymbol{I}_{M-1} \otimes \boldsymbol{B}\end{bmatrix} \quad (14)$$

$$\boldsymbol{G}(\boldsymbol{Q}) = \begin{bmatrix}\boldsymbol{A}^T(\boldsymbol{Q}_2) & \cdots & \boldsymbol{A}^T(\boldsymbol{Q}_M)\end{bmatrix}^T \quad (15)$$

$$\boldsymbol{z} = \begin{bmatrix}\boldsymbol{x}^T & \boldsymbol{t}_2^T & \cdots & \boldsymbol{t}_M^T\end{bmatrix}^T \quad (16)$$

$$\boldsymbol{W} = \mathrm{diag}(\begin{bmatrix}\boldsymbol{W}_1 & \cdots & \boldsymbol{W}_M\end{bmatrix}) . \quad (17)$$

For a known $\boldsymbol{Q}$, the weighted least squares estimates of $\boldsymbol{X}$ and $\{\boldsymbol{t}_m\}_{m=1}^M$ are obtained from

$$\hat{\boldsymbol{z}}(\boldsymbol{Q}) = \left[\boldsymbol{H}^T(\boldsymbol{Q})\boldsymbol{W}^{-2}\boldsymbol{H}(\boldsymbol{Q})\right]^{-1}\boldsymbol{H}^T(\boldsymbol{Q})\boldsymbol{W}^{-2}\boldsymbol{y} . \quad (18)$$

The constrained estimator of $\boldsymbol{Q}$, which minimises the residual sum of squares, is

$$\begin{aligned} \hat{\boldsymbol{Q}} = & \underset{\boldsymbol{Q}\in\mathbb{R}^{2(M-1)\times 2}}{\mathrm{argmax}} \ \boldsymbol{y}^T \boldsymbol{W}^{-2}\boldsymbol{H}(\boldsymbol{Q})\hat{\boldsymbol{z}}(\boldsymbol{Q}) \\ \text{s.t.} \quad & \boldsymbol{Q}_m^T \boldsymbol{Q}_m = \boldsymbol{I}_2 \quad \text{for } m = 2, \cdots, M \\ & \det(\boldsymbol{Q}_m) = 1 \quad \text{for } m = 2, \cdots, M . \end{aligned} \quad (19)$$

It is well known from generalised Procrustes analysis, that a closed-form solution to the above problem is not available unless $M = 2$ and the same weights are applied to each column of $\boldsymbol{E}_m$. In this case, a 2D eigenvalue decomposition can be used in the computation of $\boldsymbol{Q} = \boldsymbol{Q}_2$ [19]. If $M > 2$ and the same weights are applied to each column of $\boldsymbol{E}_m$, the estimates of $\boldsymbol{z}$ and $\boldsymbol{Q}$ are computed iteratively as detailed in [19] by solving a series of eigenvalue decompositions. However, since the uncertainty in the $x$- and $y$-coordinates can be far from satisfying the condition that the same weights are applied to each column of $\boldsymbol{E}_m$, we will not use the solution from [19] here. Instead, we seek to find a solution for a general weighting matrix. Such an algorithm was proposed in [20], but it seems to be very sensitive to the starting point. Specifically, the authors suggest that at least 20 random starting points should be tried out and that the unweighted solution is not suitable to use as a starting point. This is a major drawback of the algorithm, and we, therefore, suggest that something else is done. For the 2D-case, the rotation matrix can be written as

$$\boldsymbol{Q}_m(\theta_m) = \begin{bmatrix}\cos\theta_m & -\sin\theta_m \\ \sin\theta_m & \cos\theta_m\end{bmatrix} . \quad (20)$$

Thus, the complete problem in (19) has $M - 1$ nonlinear parameters. In the case of many sensor arrays, it might be computationally very intensive to optimise such a high-dimensional nonlinear objective, so we instead attack the problem as it is traditionally solved in generalised orthogonal Procrustes analysis. That is, we iterate between estimating $\boldsymbol{X}$ and $\boldsymbol{Q}$. The main advantage of this approach is that the estimation of $\boldsymbol{Q}$ given $\boldsymbol{X}$ decouples into $M - 1$ individual 1D nonlinear optimisation problems instead of the high-dimensional problem in (19). Specifically, we have to solve problems of the form

$$\begin{aligned} \hat{\theta}_m = & \underset{\theta_m \in [-\pi, \pi)}{\mathrm{argmin}} \ (\boldsymbol{y}_m - \boldsymbol{A}(\boldsymbol{Q}_m(\theta_m))\boldsymbol{x})^T \boldsymbol{W}_m^{-1} \\ & \times \left[\boldsymbol{I}_{2S} - \boldsymbol{P}_{\boldsymbol{W}_m^{-1}\boldsymbol{B}}\right]\boldsymbol{W}_m^{-1}(\boldsymbol{y}_m - \boldsymbol{A}(\boldsymbol{Q}_m(\theta_m))\boldsymbol{x}) \end{aligned} \quad (21)$$

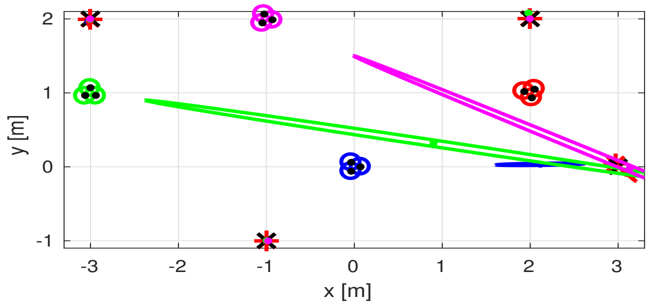


**Fig. 2**. Illustration of the quality matrices.

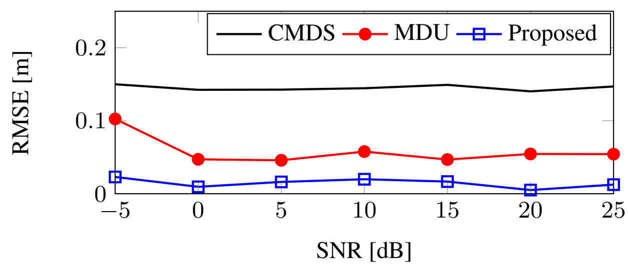


**Fig. 3**. The RMSE for estimating four sources in different SNRs and a reverberation time of 250 ms.

where $\boldsymbol{P}_{\boldsymbol{W}_m^{-1}\boldsymbol{B}} = \boldsymbol{W}_m^{-1}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{W}_m^{-2}\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{W}_m^{-1}$. In 3D, we instead get a series of 2D nonlinear optimisation problems which are not too costly to solve. Given an estimate of $\boldsymbol{Q}$, we can compute an estimate of $\boldsymbol{X}$ from (18). The algorithm can be initialised by setting the initial value of $\boldsymbol{X}$ equal to the observation matrix with the best quality matrix.

## 5. EXPERIMENTS

In this section, we present the results from three experiments. First, we illustrate how the quality matrices allow us to combine estimates having very different estimation errors. Second, we evaluate the estimation accuracy as a function of the noise level. And third, we evaluate the estimation accuracy as a function of the reverberation time. All experiments were run using MATLAB, and the code will be available at `http://tinyurl.com/jknvbn`.

In the first experiment, we used four sensor arrays, three synchronised sources, and one unsynchronised source. Each sensor array was a UCA with three microphones and a radius of 0.06 m. The calibration signal was 500 ms of white Gaussian noise which was bandpass filtered from 500 Hz to 1500 Hz. The filtering is performed since real-world loudspeakers have a large group-delay at low frequencies and are very directional at high frequencies. The sampling frequency was 4 kHz and white Gaussian noise was added so that the microphone recordings had an SNR of 10 dB. Fig. 2 shows the results. The true source and sensor positions are marked with black crosses and dots, respectively, and the source position estimates are marked with red stars. The small coloured circles denote the estimated sensor positions, and the coloured ellipses denote a contour of the quality matrices centred on the individual location estimates. For the synchronised sources, the ellipse contours are so small that they are hardly visible in the figure. For the unsynchronised sources, however, the contours indicate that the range estimates are much more uncertain than the angle estimates.

To the best of our knowledge, no other method exists which can directly solve a problem such as the one in the first experiment. Only special cases have been considered so far in the literature, and in the second experiment, we looked at one such special case treated in [13]. Specifically, we considered the case where four sensor arrays and synchronised sources were used. The sources and sensor arrays were co-located meaning that each sensor array knew the position of its own source with a very high precision. The sensor arrays were placed at the coordinates $(1,1)$, $(2.5,1)$, $(3,3)$, and $(1,2.5)$ in a room of size $(5,6,3)$ m. We computed the estimation accuracy of the sources as a function of the SNR for a reverberation time of 250 ms. The reverberation was added using a RIR-generator [21]. The proposed method was compared to two different reference methods which is a variation of [13] and the multidimensional unfolding method (MDU) in [7, 8]. The former consists in that we use the source localisation method of the proposed method to compute the local maps and classical multidimensional scaling for combining these maps. Using the same source localisation algorithm in the first reference method and the proposed method ensured that we evaluated the effect of using the proposed fitting method. For the second reference method [7, 8], we included all prior knowledge about the local geometry of the sensor arrays. As a performance measure, we used the sum of squared errors which is the dissimilarity measure often used in Procrustes analysis. For each SNR, 100 Monte Carlo runs were conducted. In each run, a new noise vector and a small random perturbation of the sensor array positions were generated. The results are shown in Fig. 3. The proposed method outperformed the reference methods across all SNRs. This demonstrates the importance of using weighting matrices, even when no unsynchronised sources are present.

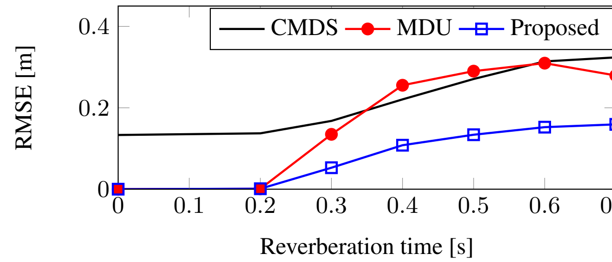


**Fig. 4**. The RMSE for estimating four sources for different reverberation times in an SNR of 20 dB.

In the third and final experiment, we had the same experimental setup as in the second experiment, except for that we varied the reverberation time and fixed the SNR to 20 dB. The results are given in Fig. 4, and they again show that the proposed method outperformed the reference methods.

## 6. CONCLUSION

In this paper, we have proposed a new two-step method for calibrating an audio system including one or several smart speakers. The method consists of a source localisation step in which each smart speaker computes a local map over the synchronised sources (e.g., loudspeakers) and unsynchronised sources (e.g., listeners). These local maps are then transmitted to a central unit which combines them into a global map in a fitting step. The fitting is performed according to the quality matrices pertaining to each local map, and they ensure that the most accurate estimates receive the greatest weight in the fitting. Via simulations, we demonstrated that the proposed method outperformed two reference methods for various noise levels and reverberation times.

## 7. REFERENCES


[1] International Telecommunication Union, Geneva, Switzerland, "Recommendation ITU-R BS.775-3, Multichannel stereophonic sound system with and without accompanying picture," 2012.

[2] J. Herre, J. Hilpert, A. Kuntz, and J. Plogsties, "MPEG-H 3D audio — the new standard for coding of immersive spatial audio," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 5, pp. 770–779, 2015.

[3] M. Crocco, A. Del Bue, and V. Murino, "A bilinear approach to the position self-calibration of multiple sensors," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 660–673, 2012.

[4] Y. Kuang, S. Burgess, A. Torstensson, and K. Astrom, "A complete characterization and solution to the microphone position self-calibration problem," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2013, pp. 3875–3879.

[5] N. D. Gaubitch, W. B. Kleijn, and R. Heusdens, "Auto-localization in ad-hoc microphone arrays," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2013, pp. 106–110.

[6] N. D. Gaubitch, W. B. Kleijn, and R. Heusdens, "Calibration of distributed sound acquisition systems using TOA measurements from a moving acoustic source," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2014, pp. 7455–7459.

[7] I. Dokmanic, J. Ranieri, and M. Vetterli, "Relax and unfold: Microphone localization with euclidean distance matrices," in *Proc. European Signal Processing Conf.*, 2015.

[8] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean distance matrices: A short walk through theory, algorithms, and applications," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 12–30, 2015.

[9] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," *IEEE Signal Process. Mag.*, vol. 33, no. 4, pp. 14–29, 2016.

[10] A. Plinge and G. Fink, "Geometry calibration of multiple microphone arrays in highly reverberant environments," in *Proc. Intl. Workshop Acoust. Echo Noise Control*. IEEE, 2014, pp. 243–247.

[11] A. Plinge, G. A. Fink, and S. Gannot, "Passive online geometry calibration of acoustic sensor networks," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 324–328, 2017.

[12] S. D. Valente, M. Tagliasacchi, F. Antonacci, P. Bestagini, A. Sarti, and S. Tubaro, "Geometric calibration of distributed microphone arrays from acoustic source correspondences," in *Proc. Int. Workshop on Multimedia Signal Process.* IEEE, 2010, pp. 13–18.

[13] P. Pertilä, M. Mieskolainen, and M. S. Hämäläinen, "Closed-form self-localization of asynchronous microphone arrays," in *Joint Workshop on Hands-free Speech Commun. and Microphone Arrays*. IEEE, 2011, pp. 139–144.

[14] J. R. Jensen, J. K. Nielsen, M. G. Christensen, and S. H. Jensen, "On frequency domain models for TDOA estimation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015.

[15] Ü. Baysal and R. L. Moses, "On the geometry of isotropic arrays," *IEEE Trans. Signal Process.*, vol. 51, no. 6, pp. 1469–1478, 2003.

[16] U. Oktel and R. L. Moses, "Source localization with isotropic arrays," *IEEE Signal Process. Lett.*, vol. 11, no. 5, pp. 501–504, 2004.

[17] J. K. Nielsen, T. L. Jensen, J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Grid size selection for nonlinear least-squares optimisation in spectral estimation and array processing," in *Proc. European Signal Processing Conf.*, 2016.

[18] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*, Englewood Cliffs, NJ, USA: Prentice Hall PTR, Mar. 1993.

[19] F. Crosilla and A. Beinat, "Use of generalised procrustes analysis for the photogrammetric block adjustment by independent models," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 56, no. 3, pp. 195–209, 2002.

[20] M. A. Koschat and D. F. Swayne, "A weighted Procrustes criterion," *Psychometrika*, vol. 56, no. 2, pp. 229–239, 1991.

[21] E. A. P. Habets, "Room impulse response generator," 2010, Ver. 2.0.20100920.