Eigenbeam-based Acoustic Source Tracking in Noisy Reverberant Environments

Daniel P. Jarrett, Emanuël A. P. Habets, Patrick A. Naylor
Department of Electrical & Electronic Engineering, Imperial College London, UK
Email: {daniel.jarrett05, e.habets, p.naylor}@imperial.ac.uk

Abstract—**In this work, an adaptive acoustic source tracking algorithm is proposed. It is based on eigenbeams which perform spatial decomposition of the sound field, works in two dimensions for tracking of azimuth and elevation, has low computational complexity, and is robust to noise and room reverberation. The tracking is performed using an adaptive principal component analysis of the particle velocity vector, which points from the acoustic source to the sensor. The particle velocity vector is estimated using a spherical microphone array, and is formed by combining the first-order eigenbeams.**

I. Introduction

Spherical microphone arrays have recently become the subject of research due to their ability to analyze 3D sound fields with high spatial resolution. Applications include spatial sound recording, direction of arrival (DOA) estimation and acoustic source tracking. Source tracking in particular is necessary for automated camera steering and microphone array beamformer steering.

There has been extensive work on localization/DOA estimation in one dimension, which has resulted in a number of different solutions: time difference of arrival (TDOA) methods such as GCC-PHAT, subspace methods such as ESPRIT and MUSIC, and steered response power methods. While some have been generalized to two dimensions [1], [2], they are typically computationally complex, due to the need for an exhaustive search, and not robust to reverberation. Additionally TDOA-based methods are unsuitable for practical spherical microphone arrays with a small radius, due to the insufficient spacing between microphones.

In this paper we propose a source tracking algorithm based on a low computational complexity DOA estimation method. The low complexity of this method is crucial for real-time tracking applications. The tracking is performed using an adaptive principal component analysis of the acoustic particle velocity vector, and is robust to noise and reverberation.

In previous work [3] we drew upon the theory of spherical harmonics and proposed a low complexity DOA estimation method based on eigenbeams, a spatial (in this case spherical) decomposition of the sound field. These eigenbeams were used to compute a pseudointensity vector, which is similar to the intensity vector in acoustics, and points from the acoustic source to the receiver. In this work we use a similar eigenbeam-based method to estimate the particle velocity vector, which can also be used for DOA estimation [4]. This vector points from the acoustic source to the receiver for plane wave incidence; we therefore assume far-field conditions, a point source and a point sensor.

This paper is organized as follows: in Section II we introduce the particle velocity vector and formulate the DOA estimation problem as a maximum SNR beamforming problem, in Section III we explain our method for estimating the particle velocity vector using a spherical microphone array, in Section IV we present our adaptive tracking algorithm, and in Section V we evaluate its performance in simulated environments with different noise and reverberation levels.

II. Problem formulation

A. Particle velocity vector

If $\mathbf{u}$ is a unit vector pointing from the sensor towards an acoustic source and we assume a plane wave at the sensor, the acoustic particle velocity vector $\mathbf{s}$ at a time instant n is given by [5, p. 31]:

$$\mathbf{s}(n) = -\frac{p(n)}{Z_0}\mathbf{u}(n), \tag{1}$$

where p is the sound pressure and Z_0 is the characteristic acoustic impedance of air. A spherical microphone array allows us to construct an estimate of the particle velocity vector $\mathbf{s}$; our task is then to estimate the source location, i.e. the vector $\mathbf{u}$.

B. Maximum SNR beamforming

Let $\mathbf{v}(n) = [v_x(n), v_y(n), v_z(n)]^T$ be the noisy input signal, a time-varying particle velocity vector. The noise is modelled by a term $\mathbf{e}(n) = [e_x(n), e_y(n), e_z(n)]^T$ which can include both ambient noise and room reverberation. The desired and noise signals are assumed to be mutually independent; the reflections due to reverberation are therefore assumed to be diffuse. We have:

$$\begin{aligned}\mathbf{v}(n) &= \mathbf{s}(n) + \mathbf{e}(n)\\ &= -\frac{p(n)}{Z_0}\,\mathbf{u}(n) + \mathbf{e}(n),\end{aligned} \tag{2}$$

If we apply a time-varying spatial weighting vector $\mathbf{w}(n)$ to the input signal $\mathbf{v}(n)$, and sum the resulting three signals, we get an output signal $y(n)$:

$$\begin{aligned}y(n) &= \mathbf{w}^T(n)\,\mathbf{v}(n)\\ &= \mathbf{w}^T(n)\,\mathbf{s}(n) + \mathbf{w}^T(n)\,\mathbf{e}(n)\\ &= -\frac{p(n)}{Z_0}\mathbf{w}^T(n)\,\mathbf{u}(n) + \mathbf{w}^T(n)\mathbf{e}(n).\end{aligned} \tag{3}$$

The output signal-to-noise ratio (SNR) can be defined as

$$\begin{aligned}\mathrm{oSNR}(\mathbf{w}) &= \frac{\mathbf{w}^T\mathbf{R}_{ss}\mathbf{w}}{\mathbf{w}^T\mathbf{R}_{ee}\mathbf{w}}\\ &= \frac{\sigma_s^2\left(\mathbf{w}^T\mathbf{u}\right)^2}{\mathbf{w}^T\mathbf{R}_{ee}\mathbf{w}},\end{aligned} \tag{4}$$

where $\mathbf{R}_{ss} = E\{\mathbf{s}(n)\mathbf{s}^T(n)\}$ is the covariance matrix of the desired source signal and $\mathbf{R}_{ee} = E\{\mathbf{e}(n)\mathbf{e}^T(n)\}$ is the covariance matrix of the noise. As the desired signal $\mathbf{s}$ and the noise signal $\mathbf{e}$ are mutually independent, the covariance matrix of the input signal $\mathbf{v}$ can be expressed as $\mathbf{R}_{vv} = \mathbf{R}_{ss} + \mathbf{R}_{ee}$ and we can express the variance of the output y as

$$\begin{aligned}\sigma_y^2 &= \mathbf{w}^T\mathbf{R}_{vv}\mathbf{w}\\ &= \mathbf{w}^T\left[\mathbf{R}_{ss} + \mathbf{R}_{ee}\right]\mathbf{w}\\ &= \mathbf{w}^T\mathbf{R}_{ss}\mathbf{w} + \mathbf{w}^T\mathbf{R}_{ee}\mathbf{w}.\end{aligned} \tag{5}$$

The beamformer with weights $\mathbf{w}$ that maximizes the output SNR $\mathrm{oSNR}(\mathbf{w})$ is known as a maximum SNR beamformer. This is equivalent to determining the principal component of the data set comprising the noisy observations of the particle velocity vector.

Let us now assume spherically white noise such that

$$\mathbf{R}_{ee} = \sigma_e^2\mathbf{I}, \tag{6}$$

where $\mathbf{I}$ denotes a 3×3 identity matrix. Substituting this expression in (5), one can see that maximising the output SNR in (4) is equivalent to maximizing the power of $y(n)$ under the constraint

$$\mathbf{w}^T\mathbf{w} = 1. \tag{7}$$

Therefore, our objective can be formulated as

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} \mathbf{w}^T\mathbf{R}_{vv}\mathbf{w} \quad \text{s.t.} \quad \mathbf{w}^T\mathbf{w} = 1. \tag{8}$$

The optimal solution $\mathbf{w}_{\mathrm{o}}$ is given by $\mathbf{u} = \mathbf{s}/\|\mathbf{s}\|$, where $||\cdot||$ denotes the l-2 norm.

For the more general problem where the noise is not spherically white, the objective function would be given by

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} \mathbf{w}^T\mathbf{R}_{vv}\mathbf{w} \quad \text{s.t.} \quad \mathbf{w}^T\mathbf{R}_{ee}\mathbf{w} = 1. \tag{9}$$

In this case an estimate of $\mathbf{R}_{ee}$ would be required.

III. Eigenbeam-based particle velocity vector estimation

The noisy particle velocity vector $\mathbf{v}$ can be measured using an acoustic vector sensor (e.g. the Microflown [6]), however here we wish to measure it using conventional pressure sensors. In [3], we proposed a method for estimating the acoustic intensity vector using eigenbeams, which are the result of the decomposition of a three-dimensional sound pressure field into a set of eigensolutions of the acoustic wave equation in spherical polar coordinates. These eigenbeams can be obtained using a spherical microphone array. Since the direction of the intensity vector $\mathbf{I} = p\mathbf{v}$ is given by the particle velocity vector, we can use this method here to estimate the particle velocity vector.

A. Eigenbeam calculation

Let us consider a sound pressure field $p(k, \mathbf{r})$ at a point $\mathbf{r} = (r, \Omega) = (r, \theta, \phi)$ (in spherical polar coordinates, with elevation θ and azimuth ϕ) on a sphere, where k is the wavenumber. The spherical Fourier transform of this field is given by [7, p. 192]:

$$p_{lm}(k) = \int_{\Omega\in S^2} p(k, r, \Omega) Y_{lm}^*(\Omega) d\Omega, \tag{10}$$

where $(\cdot)^*$ denotes the complex conjugate and $\int_{\Omega\in S^2} d\Omega \triangleq \int_0^{2\pi}\int_0^{\pi} \sin\theta d\theta d\phi$. We call p_{lm} the *eigenbeam* of order l and degree m. Our basis functions are *spherical harmonics* $Y_{lm}(\Omega)$ given by [7, p. 190]:

$$Y_{lm}(\Omega) = \sqrt{\frac{(2l+1)}{4\pi}\frac{(l-m)!}{(l+m)!}} P_{lm}(\cos\theta) e^{im\phi}, \tag{11}$$

where P_{lm} is the associated Legendre function and $i = \sqrt{-1}$.

With a spherical microphone array we do not have a continuous pressure sensor, but instead a sensor formed from discretely spaced microphones. If we have a finite number of microphones Q with spherical polar coordinates $\mathbf{r_q} = (r_q, \Omega_q), q = 1, \dots, Q$, we can approximate the integral in (10) with a weighted sum of the pressure at each microphone [8]:

$$p_{lm}(k) \approx \sum_{q=1}^{Q} p(k, r_q, \Omega_q)\, g_{q,lm}. \tag{12}$$

The weights $g_{q,lm}$, which depend on the sampling configuration, must be chosen such that (12) is an accurate approximation of (10). As the spherical harmonics are orthonormal [9], choosing

$$g_{q,lm} = \frac{4\pi}{Q} Y_{lm}^*(\Omega_q)$$

makes this approximation exact provided the microphones are equally spaced on the sphere and $Q \geq (N+1)^2$, where N is the highest harmonic order. In practice a small error is involved as perfectly equidistant microphone positions cannot be determined for non-trivial configurations.

B. Particle velocity vector estimation

The particle velocity vector can be estimated from the first-order ($l = 1$) eigenbeams $p_{1(-1)}(k)$, $p_{10}(k)$ and $p_{11}(k)$, as in [3]:

$$\mathbf{v}(k) = \begin{bmatrix} v_x(k) \\ v_y(k) \\ v_z(k) \end{bmatrix} \tag{13}$$

where the components $v_x(k)$, $v_y(k)$ and $v_z(k)$ of this vector are dipoles steered in the opposite direction to the x, y and z axes, to account for the minus sign in (1). This yields a vector $\mathbf{v}(k)$ that is proportional to the particle velocity vector, which is sufficient since we are only interested in its direction.

The steered beams $v_x(k)$, $v_y(k)$ and $v_z(k)$ are formed using a linear combination of rotated first-order eigenbeams [3]:

$$v_a(k) = \frac{1}{b_1(k)} \sum_{m=-1}^{1} \alpha_{a,m}\, p_{1m}(k), \quad a \in \{x, y, z\} \qquad (14)$$

where the $b_1(k)$ factor is the first-order mode strength and is required to make the beam patterns wavenumber independent.

To steer each of the eigenbeams in the appropriate direction (θ_r, ϕ_r), we multiply them by the spherical harmonics $Y_{1m}(\theta_r, \phi_r)$. We therefore require:

$$\begin{aligned}\alpha_{x,m} &= Y_{1m}(\pi/2, \pi),\\ \alpha_{y,m} &= Y_{1m}(\pi/2, -\pi/2),\\ \alpha_{z,m} &= Y_{1m}(\pi, 0).\end{aligned}$$

The particle velocities $v_a(n)$ in the discrete time domain are then obtained by taking the inverse discrete Fourier transform of the beams $v_a(k)$ evaluated at discrete values of wavenumber k.

IV. Adaptive localization algorithm

A. Gradient ascent algorithm for spherically white noise

The constraint optimization problem in (8) can be solved using the method of Lagrange multipliers:

$$L(\mathbf{w}, \lambda) = \mathbf{w}^T \mathbf{R}_{vv} \mathbf{w} + \lambda \left(\mathbf{w}^T \mathbf{w} - 1\right), \qquad (15)$$

where λ denotes the Lagrange multiplier. The update equation is given by

$$\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n{-}1) + \mu \nabla L_{\mathbf{w}}|_{\mathbf{w}=\hat{\mathbf{w}}(n-1)}, \qquad (16)$$

where μ is the step size and

$$\nabla L_{\mathbf{w}} = 2\mathbf{R}_{vv}\mathbf{w} + \lambda \mathbf{w}. \qquad (17)$$

We determine λ such that $\mathbf{w}^T(n)\mathbf{w}(n) = 1$, neglecting terms of $\mathcal{O}(\mu^2)$, as follows:

$$\left[\mathbf{w}(n{-}1) + \mu \nabla L_{\mathbf{w}}\right]^T \left[\mathbf{w}(n{-}1) + \mu \nabla L_{\mathbf{w}}\right] = 1$$

$$\mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1) + 2\mu \mathbf{w}^T(n{-}1)\left[2\mathbf{R}_{vv}\mathbf{w}(n{-}1) + \lambda \mathbf{w}(n{-}1)\right] = 1$$

$$\mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1) + 4\mu \mathbf{w}^T(n{-}1)\mathbf{R}_{vv}\mathbf{w}(n{-}1) + 2\mu\lambda \mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1) = 1$$

$$\frac{1 - \mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1) - 4\mu \mathbf{w}^T(n{-}1)\mathbf{R}_{vv}\mathbf{w}(n{-}1)}{2\mu \mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1)} = \lambda$$

$$\frac{1 - \mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1)}{2\mu \mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1)} - \frac{2\mathbf{w}^T(n{-}1)\mathbf{R}_{vv}\mathbf{w}(n{-}1)}{\mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1)} = \lambda.$$

Now we obtain the update equation by substituting λ into (16):

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \hat{\mathbf{w}}(n{-}1) + \mu \Bigg[2\mathbf{R}_{vv}\hat{\mathbf{w}}(n{-}1)\\ &+ \left(\frac{1 - \mathbf{w}^T(n{-}1)\mathbf{w}(n{-}1)}{2\mu \hat{\mathbf{w}}^T(n{-}1)\hat{\mathbf{w}}(n{-}1)} - \frac{2\hat{\mathbf{w}}^T(n{-}1)\mathbf{R}_{vv}\hat{\mathbf{w}}(n{-}1)}{\hat{\mathbf{w}}^T(n{-}1)\hat{\mathbf{w}}(n{-}1)}\right)\hat{\mathbf{w}}(n{-}1)\Bigg]\end{aligned} \qquad (18)$$

which can be written as

$$\begin{aligned}\hat{\mathbf{w}}(n) &= \frac{1}{2}\left[\frac{1}{\hat{\mathbf{w}}^T(n{-}1)\hat{\mathbf{w}}(n{-}1)} + 1\right]\hat{\mathbf{w}}(n{-}1)\\ &+ \mu\left[2\mathbf{R}_{vv}\hat{\mathbf{w}}(n{-}1) - \frac{2\hat{\mathbf{w}}^T(n{-}1)\mathbf{R}_{vv}\hat{\mathbf{w}}(n{-}1)}{\hat{\mathbf{w}}^T(n{-}1)\hat{\mathbf{w}}(n{-}1)}\hat{\mathbf{w}}(n{-}1)\right].\end{aligned} \qquad (19)$$

B. Sign ambiguity

Principal component analysis (PCA) and the method described in Section IV-A have an inherent sign ambiguity which is not mathematically solvable. To obtain an estimate $\hat{\mathbf{u}}$ of $\mathbf{u}$ which points in the correct direction, we need to determine the correct sign from an analysis of the data. This can be done by looking at the sign of the correlation r_{yp} between y and p: if it is positive, then $\mathbf{u}$ points in the opposite direction to $\mathbf{w}$, and if it is negative, then $\mathbf{u}$ points in the same direction as $\mathbf{w}$:

$$\mathbf{u}(n) = -\mathrm{sign}\left(r_{yp}\right)\mathbf{w}(n). \qquad (20)$$

C. Implementation

For an efficient implementation which allows for tracking, we do not perform the processing on a per sample basis, but instead on a frame by frame basis. We initialise the algorithm for frame $\ell = 0$ using a standard PCA, i.e., we take the eigenvector corresponding to the largest eigenvalue of the data covariance matrix $\mathbf{R}_{vv}(0)$.

Let L denote the frame length and M the frame increment, thus yielding an overlap of 75% for $M = L/4$ for example. The covariance matrix can be recursively estimated over L samples:

$$\hat{\mathbf{R}}_{vv}(\ell) = \beta_R \hat{\mathbf{R}}_{vv}(\ell{-}1) + (1 - \beta_R)\frac{1}{L}\sum_{n=\ell M}^{\ell M+L-1} \mathbf{v}(n)\mathbf{v}^T(n), \qquad (21)$$

where β_R is a weighting factor: the larger the weighting factor, the larger the contribution of previous samples. Similarly for the correlation, we have:

$$\hat{r}_{yp}(\ell) = \beta_r \hat{r}_{yp}(\ell{-}1) + (1 - \beta_r)\frac{1}{L}\sum_{n=\ell M}^{\ell M+L-1} \hat{\mathbf{w}}^T(n)\mathbf{v}(n)\,\hat{p}(n), \qquad (22)$$

where β_r is a weighting factor similar to β_R.

The update equation for $\hat{\mathbf{w}}$ is given by:

$$\begin{aligned}\hat{\mathbf{w}}(\ell) &= \frac{1}{2}\left[\frac{1}{\hat{\mathbf{w}}^T(\ell{-}1)\hat{\mathbf{w}}(\ell{-}1)} + 1\right]\hat{\mathbf{w}}(\ell{-}1)\\ &+ \mu\left[2\hat{\mathbf{R}}_{vv}(\ell)\hat{\mathbf{w}}(\ell{-}1) - \frac{2\hat{\mathbf{w}}^T(\ell{-}1)\hat{\mathbf{R}}_{vv}(\ell)\hat{\mathbf{w}}(\ell{-}1)}{\hat{\mathbf{w}}^T(\ell{-}1)\hat{\mathbf{w}}(\ell{-}1)}\hat{\mathbf{w}}(\ell{-}1)\right].\end{aligned} \qquad (23)$$

Finally the estimated unit vector pointing from the sensor towards the source, for frame ℓ, is given by:

$$\hat{\mathbf{u}}(\ell) = -\mathrm{sign}\left(\hat{r}_{yp}(\ell)\right)\hat{\mathbf{w}}(\ell). \qquad (24)$$

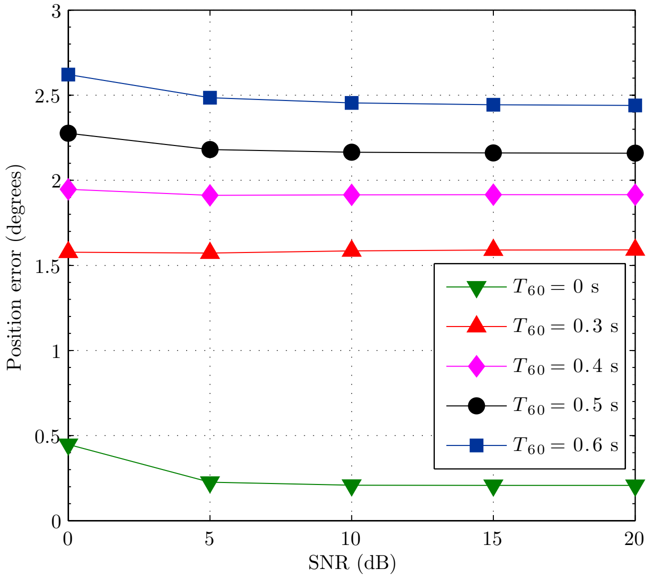

Fig. 1. Position error as a function of SNR and reverberation time T_{60}.

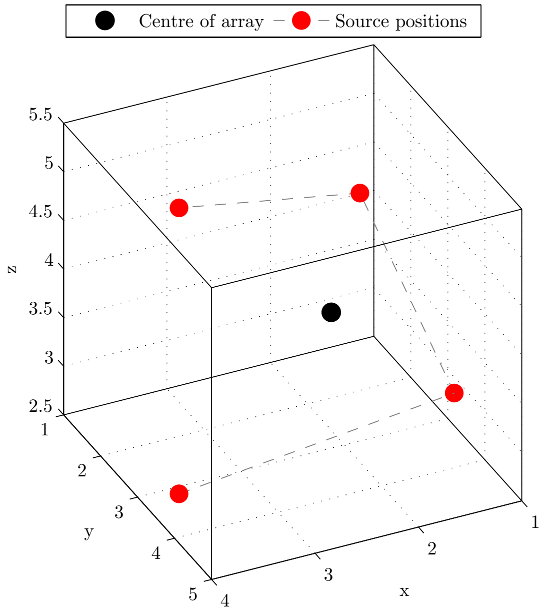

Fig. 2. Source positions relative to centre of spherical microphone array.

V. Performance evaluation

A. Experiment setup

We tested our algorithm in a room acoustics scenario simulated using SMIRgen, a recently developed room impulse response (RIR) generator for spherical microphone arrays. The receiver, a 32 microphone rigid spherical microphone array of radius 4.2 cm (the same specifications as the *Eigenmike*$^{\text{TM}}$), was placed near the centre of a simulated $4 \times 6 \times 8$ m room.

We limited the RIR to 2048 samples, with a sampling frequency of 8 kHz. The source signal was 2 s of white Gaussian noise.

B. Static source

In a first experiment for a static source, we performed Monte Carlo simulations with 10 runs, for various SNRs and room reverberation times T_{60}. For each run a new source position was randomly selected, at a distance of 1.5 m from the centre of the array. We chose a step size $\mu = 1$, weighting factors $\beta_R = 0.95$ and $\beta_r = 0.98$, frame length $L = 256$ and frame increment $M = 64$.

To evaluate the performance of our algorithm we computed the angular error ϵ, which is the angle between a unit vector $\mathbf{u}$ pointing in the correct direction and a unit vector $\hat{\mathbf{u}}$ pointing in the estimated direction. As these are unit vectors, ϵ is given by:

$$\epsilon = \cos^{-1}(\mathbf{u}^T \hat{\mathbf{u}}) \tag{25}$$

The angular error averaged over all estimates from 1.5 to 2 s is shown in Fig. 1. It can be seen that even with reverberation times up to 600 ms and SNRs as low as 0 dB, the angular error remains below 3.5°.

C. Moving source

In a second experiment, over a period of 2 s we placed the source in four different positions around the array, at a distance of 1.5-2 m from the centre of the array, as illustrated in Fig. 2. We chose $\mu = 0.3$, $\beta_R = 0.9$, $\beta_r = 0.95$, $L = 128$ and $M = 32$.

The reference and estimated source positions are shown in Fig. 3 for various reverberation times and an SNR of 5 dB. After an initial tracking time, the estimates converge to the true position, within a couple of degrees. The results are similar for SNRs above 5 dB. While the tracking time generally increases as the reverberation time increases, after tracking the accuracy of the estimates is good even for high reverberation times. It should be noted that while in some cases it appears the estimate is diverging from the true position (e.g. at 500–700 ms), this is due to the sign ambiguity: once the sign has changed (e.g. at 600 ms), it can be seen that the estimate is actually converging towards the true position.

D. Choice of adaptive parameters

If we wish to reduce the tracking time, we can increase μ and decrease β_R and β_r, at the risk of creating instability and at the expense of accuracy. If we wish to increase the accuracy, we can increase β_R and β_r and decrease μ, at the expense of a higher tracking time.

VI. Conclusion

The proposed algorithm allows us to track sources in two dimensions (azimuth and elevation) using a spherical microphone array. An evaluation of this algorithm has shown that it has high accuracy, even in the presence of high levels of noise, down to SNRs of 0-5 dB, and reverberation times up to 600 ms.

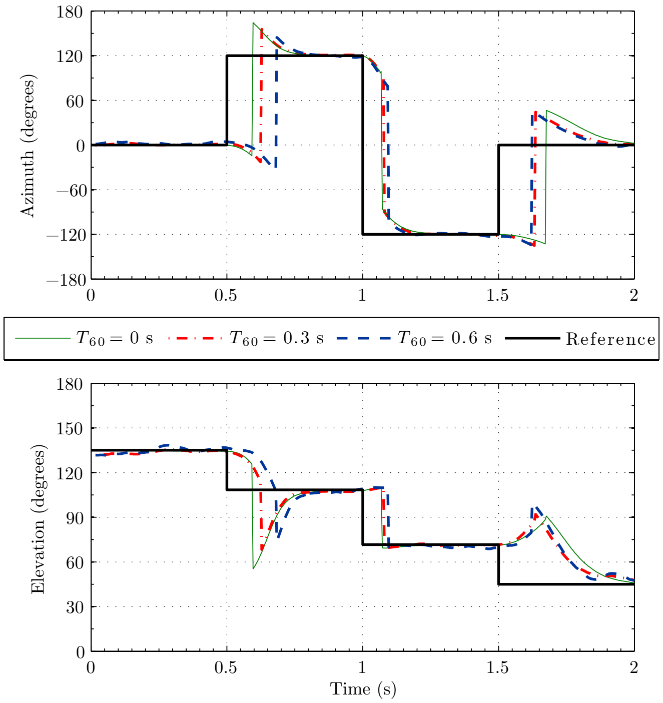

Fig. 3. Reference and estimated source positions as a function of time, for various reverberation times T_{60}.

Acknowledgment

This research was supported by the Marie Curie Intra-European Fellowship and the Future and Emerging Technologies Programme within the Seventh European Community Framework Programme under contract number PIEF-GA-2009-237246 and FET-Open grant number 226007 SCENIC, respectively.

References

[1] D. Khaykin and B. Rafaely, "Coherent signals direction-of-arrival estimation using a spherical microphone array: Frequency smoothing approach," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2009, pp. 221–224.

[2] H. Teutsch and W. Kellermann, "Eigen-beam processing for direction-of-arrival estimation using spherical apertures," in *Proc. Joint Workshop on Hands-Free Speech Communication and Microphone Arrays*. Piscataway, New Jersey, USA: IEEE, Mar. 2005.

[3] D. P. Jarrett, E. A. P. Habets, and P. A. Naylor, "3D source localization in the spherical harmonic domain using a pseudointensity vector," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Aalborg, Denmark, Aug. 2010.

[4] A. Nehorai and E. Paldi, "Acoustic vector-sensor array processing," *IEEE Trans. Signal Process.*, vol. 42, no. 9, pp. 2481–2491, Sep. 1994.

[5] M. J. Crocker, Ed., *Handbook of Acoustics*. Wiley-Interscience, 1998.

[6] H.-E. de Bree, P. Leussink, T. Korthorst, H. Jansen, T. S. Lammerink, and M. Elwenspoek, "The μ-flown: a novel device for measuring acoustic flows," vol. 54, no. 1-3. Elsevier, 1996, pp. 552–557.

[7] E. G. Williams, *Fourier Acoustics: Sound Radiation and Nearfield Acoustical Holography*, 1st ed. Academic Press, 1999.

[8] B. Rafaely, B. Weiss, and E. Bachmat, "Spatial aliasing in spherical microphone arrays," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 1003–1010, Mar. 2007.

[9] J. Meyer and G. Elko, "A highly scalable spherical microphone array based on an orthonormal decomposition of the soundfield," in *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, May 2002, pp. 1781–1784.