Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2007, Article ID 50870, 11 pages
doi:10.1155/2007/50870

*Research Article*

# Particle Filter with Integrated Voice Activity Detection for Acoustic Source Tracking

**Eric A. Lehmann and Anders M. Johansson**

*Western Australian Telecommunications Research Institute, 35 Stirling Highway, Perth, WA 6009, Australia*

Received 28 February 2006; Revised 1 August 2006; Accepted 26 August 2006

Recommended by Joe C. Chen

In noisy and reverberant environments, the problem of acoustic source localisation and tracking (ASLT) using an array of microphones presents a number of challenging difficulties. One of the main issues when considering real-world situations involving human speakers is the temporally discontinuous nature of speech signals: the presence of silence gaps in the speech can easily misguide the tracking algorithm, even in practical environments with low to moderate noise and reverberation levels. A natural extension of currently available sound source tracking algorithms is the integration of a voice activity detection (VAD) scheme. We describe a new ASLT algorithm based on a particle filtering (PF) approach, where VAD measurements are fused within the statistical framework of the PF implementation. Tracking accuracy results for the proposed method is presented on the basis of synthetic audio samples generated with the image method, whereas performance results obtained with a real-time implementation of the algorithm, and using real audio data recorded in a reverberant room, are published elsewhere. Compared to a previously proposed PF algorithm, the experimental results demonstrate the improved robustness of the method described in this work when tracking sources emitting real-world speech signals, which typically involve significant silence gaps between utterances.



## 1. INTRODUCTION

The concept of speaker localisation and tracking using an array of acoustic sensors has become an increasingly important field of research over the last few years [1–3]. Typical applications such as teleconferencing, automated multi-media capture, smart meeting rooms and lecture theatres, and so forth, are fast becoming an engineering reality. This in turn requires the development of increasingly sophisticated algorithms to deal efficiently with problems related to background noise and acoustic reverberation during the audio data acquisition process.

A major part of the literature on the specific topic of acoustic source localisation and tracking (ASLT) typically focuses on implementations involving human speakers [1–9]. One of the major difficulties in a practical implementation of ASLT for speech-based applications lies in the non-stationary character of typical speech signals, with potentially significant silence periods existing between separate utterances. During such silence gaps, currently available ASLT methods will usually keep updating the source location estimates as if the speaker was still active. The algorithm is therefore likely to momentarily lose track of the true source position since the updates are then based solely on disturbance sources such as reverberation and background noise, whose influence might be quite significant in practical situations. Whether the algorithm recovers from this momentary tracking error or not, and how fast the recovery process occurs, is mainly determined by how long the silence gap lasts. Consequently, existing works on acoustic source tracking either implicitly rely on the fact that silence periods in the considered speech signal remain relatively short [2–5], or alternatively, assume a stationary source signal, as in vehicle tracking applications for instance [10, 11].

In the present work, we address this specific problem by presenting a new algorithm for ASLT that includes the data obtained from a voice activity detector (VAD) as an integral part of the target-tracking process. To the best of our knowledge, this fusion problem is yet to be considered in the acoustic source tracking literature, despite the fact that this approach can be regarded as a natural extension of currently existing ASLT algorithms developed for speech-based applications. In this paper, we use an approach based on a particle filtering (PF) concept similar to that used previously in [2], and show how the VAD measurement modality can be efficiently fused within the statistical framework of sequential

Monte Carlo (SMC) methods. Rather than simply using this additional measurement in the derivation of a mixed-mode likelihood, we consider the VAD data as a prior probability that the source localisation observations originate from the true source. As a result, the proposed particle filter, denoted PF-VAD, integrates the VAD data at a low level in the PF algorithm development. It hence benefits from the various advantages inherent to SMC methods (nonlinear and non-Gaussian processing) and is able to deal efficiently with significant gaps in the speech signal.

This paper is organised as follows. The next section first provides a generic definition of the considered tracking problem, and then briefly reviews the basic principles of Bayesian filtering (state-space approach). In Section 3, we derive the theoretical concepts required by the PF methodology on the basis of the specific ASLT problem definition; the derivation of this statistical framework then allows the integration of VAD measurements within the PF algorithm. Section 4 contains a review of the VAD scheme used in this work (based on [12]), and we then update this basic scheme for the specific speaker tracking purpose considered in this work. We further derive three different types of VAD outputs (considering both hard and soft decisions) to be used within the PF algorithm, and the proposed PF-VAD method is finally presented in Section 5. A performance assessment of this algorithm is then given in Section 6, which also includes the results obtained with a PF method previously developed in [2] for comparison purposes. The paper finally concludes with a summary of the results and some future work considerations in Section 7.

## 2. BAYESIAN FILTERING FOR TARGET TRACKING

### 2.1. ASLT problem definition

Consider an array of $M$ acoustic sensors distributed at known locations in a reverberant environment with known acoustic wave propagation speed $c$. For a typical application of speaker tracking, the microphones are usually scattered around the considered enclosure in such a way that the acoustic source always remains within the interior of the sensor array. This type of setup allows for a better localisation accuracy compared to, for instance, a concentrated linear or circular array. Assuming a single sound source, the problem consists in estimating the location of this "target" in the current coordinate system based on the signals $f_m(t)$, $m \in \{1,\ldots,M\}$, provided by the microphones. It is further assumed that the sensor signals are sampled in time and decomposed into a series of successive frames $k = 1,2,\ldots$, of equal length $L$ before being processed. The problem is then considered on the basis of the discrete-time variable $k$.

Note that the derivations presented in this work focus on a two-dimensional problem setting where the height of the source is considered known, or of no particular importance. The acoustic sensors are therefore placed at a constant height in the enclosure, and the aim is to ultimately provide a two-dimensional estimate of the source location on this horizontal plane only. The following developments can however be easily generalised to include the third dimension if necessary.

### 2.2. State-space filtering

Assuming that a Cartesian coordinate system with known origin has been defined for the considered tracking problem, let $\mathbf{X}_k$ represent the state variable for time frame $k$, corresponding to the position $[x_k \;\; y_k]^{\mathrm{T}}$ and velocity $[\dot{x}_k \;\; \dot{y}_k]^{\mathrm{T}}$ of the target in the state space:

$$\mathbf{X}_k = \begin{bmatrix} x_k & y_k & \dot{x}_k & \dot{y}_k \end{bmatrix}^{\mathrm{T}}. \tag{1}$$

At any time step $k$, each microphone in the array delivers a frame of audio signal which can be processed using some localisation technique such as, for instance, steered beamforming (SBF) or time-delay estimation (TDE). Let $\mathbf{Y}_k$ denote the observation variable (measurement) which, in the case of ASLT, typically corresponds to the localisation information resulting from this preprocessing of the audio signals.

Using a Bayesian filtering approach and assuming Markovian dynamics, this system can be globally represented by means of the following two equations [13]:

$$\mathbf{X}_k = g(\mathbf{X}_{k-1}, \mathbf{u}_k), \tag{2a}$$

$$\mathbf{Y}_k = h(\mathbf{X}_k, \mathbf{v}_k), \tag{2b}$$

where $g(\cdot)$ and $h(\cdot)$ are possibly nonlinear functions, and $\mathbf{u}_k$ and $\mathbf{v}_k$ are possibly non-Gaussian noise variables. Ultimately, one would like to compute the so-called posterior probability density function (PDF) $p(\mathbf{X}_k \mid \mathbf{Y}_{1:k})$, where $\mathbf{Y}_{1:k} = \{\mathbf{Y}_1,\ldots,\mathbf{Y}_k\}$ represents the concatenation of all measurements up to time $k$. The density $p(\mathbf{X}_k \mid \mathbf{Y}_{1:k})$ contains all the statistical information available regarding the current condition of the state variable $\mathbf{X}_k$, and an estimate $\hat{\mathbf{X}}_k$ of the state then follows, for instance, as the mean or the mode of this PDF.

The solution to this Bayesian filtering problem consists of the following two steps of prediction and update [14]. Assuming that the posterior density $p(\mathbf{X}_{k-1} \mid \mathbf{Y}_{1:k-1})$ is known at time $k-1$, the posterior PDF $p(\mathbf{X}_k \mid \mathbf{Y}_{1:k})$ for the current time step $k$ can be computed using the following equations:

$$\begin{aligned} p(\mathbf{X}_k \mid \mathbf{Y}_{1:k-1}) &= \int p(\mathbf{X}_k \mid \mathbf{X}_{k-1})\, p(\mathbf{X}_{k-1} \mid \mathbf{Y}_{1:k-1})\, \mathrm{d}\mathbf{X}_{k-1}, \\ p(\mathbf{X}_k \mid \mathbf{Y}_{1:k}) &\propto p(\mathbf{Y}_k \mid \mathbf{X}_k)\, p(\mathbf{X}_k \mid \mathbf{Y}_{1:k-1}), \end{aligned} \tag{3}$$

where $p(\mathbf{X}_k \mid \mathbf{X}_{k-1})$ is the transition density, and $p(\mathbf{Y}_k \mid \mathbf{X}_k)$ is the so-called likelihood function.

### 2.3. Sequential Monte Carlo (SMC) approach

Particle filtering (PF) is an approximation technique that solves the Bayesian filtering problem by representing the posterior density as a set of $N$ samples of the state space $\mathbf{X}_k^{(n)}$ (particles) with associated weights $w_k^{(n)}$, $n \in \{1,\ldots,N\}$, see, for example, [14]. The implementation of SMC methods represents a powerful tool in the sense that they can be efficiently applied to nonlinear and/or non-Gaussian problems, contrary to other approaches such as the Kalman filter and

its derivatives. Originally proposed by Gordon et al. [15], the so-called bootstrap algorithm is an attractive PF variant due to its simplicity of implementation and low computational demands. Assuming that the set of particles and weights $\{(\mathbf{X}_{k-1}^{(n)}, w_{k-1}^{(n)})\}_{n=1}^{N}$ is a discrete representation of the posterior density at time $k-1$, $p(\mathbf{X}_{k-1} \mid \mathbf{Y}_{1:k-1})$, the generic iteration update for the bootstrap PF algorithm is given in Algorithm 1. Following this iteration, the new set of particles and weights $\{(\mathbf{X}_k^{(n)}, w_k^{(n)})\}_{n=1}^{N}$ is approximately distributed as the current posterior density $p(\mathbf{X}_k \mid \mathbf{Y}_{1:k})$. The sample set approximation of the posterior PDF can then be obtained using

$$p(\mathbf{X}_k \mid \mathbf{Y}_{1:k}) \approx \sum_{n=1}^{N} w_k^{(n)} \delta\left(\mathbf{X}_k - \mathbf{X}_k^{(n)}\right), \tag{4}$$

where $\delta(\cdot)$ is the Dirac delta function, and an estimate $\hat{\mathbf{X}}_k$ of the target state for the current time step $k$ follows as

$$\hat{\mathbf{X}}_k = \int \mathbf{X}_k \cdot p(\mathbf{X}_k \mid \mathbf{Y}_{1:k})\, \mathrm{d}\mathbf{X}_k \tag{5a}$$

$$\approx \sum_{n=1}^{N} w_k^{(n)} \mathbf{X}_k^{(n)}. \tag{5b}$$

It can be shown that the variance of the weights $w_k^{(n)}$ can only increase over time, which decreases the overall accuracy of the algorithm. This constitutes the so-called degeneracy problem, known to affect PF implementations. The conditional resampling step in Algorithm 1 is introduced as way to mitigate these effects. This resampling process can be easily implemented using a scheme based on a cumulative weight function, see, for example, [15]. Alternatively, several other resampling methods are also available from the particle filtering literature [14].

The main disadvantage of the bootstrap algorithm is that during the prediction step, the particles are relocated in the state space without knowledge of the current measurement $\mathbf{Y}_k$. Some regions of the state space with potentially high posterior likelihood might hence be omitted during the iteration. Despite this drawback, this algorithm constitutes a good basis for the evaluation of particle filtering methods in the context of the current application, keeping in mind that the use of a more elaborate PF method would also increase the accuracy of the resulting tracking algorithm.

## 3. PF FOR ACOUSTIC SOURCE TRACKING

The particle filtering concepts presented in this section are based upon those derived previously in [2], where a sequential estimation framework was developed for the specific problem of acoustic source localisation and tracking. More information on this topic can be found in this publication and the references cited therein if necessary.

From Algorithm 1, it can be seen that the particle filtering method involves the definition of two important concepts: the source dynamics (through the transition function $g(\cdot)$) and the likelihood function $p(\mathbf{Y}_k \mid \mathbf{X}_k)$, which are derived in the sequel.

**Assumption:** at time $k-1$, the set of particles $\mathbf{X}_{k-1}^{(n)}$ and weights $w_{k-1}^{(n)}$, $n \in \{1, \ldots, N\}$, is a discrete representation of the posterior $p(\mathbf{X}_{k-1} \mid \mathbf{Y}_{1:k-1})$.

**Iteration:** given the observation $\mathbf{Y}_k$ obtained at the current time $k$, update the particle set as follows:

(1) *Prediction:* propagate the particles through the transition equation, $\tilde{\mathbf{X}}_k^{(n)} = g(\mathbf{X}_{k-1}^{(n)}, \mathbf{u}_k)$.

(2) *Update:* assign each particle a likelihood weight, $\tilde{w}_k^{(n)} = w_{k-1}^{(n)} \cdot p(\mathbf{Y}_k \mid \tilde{\mathbf{X}}_k^{(n)})$, then normalize the weights:

$$w_k^{(n)} = \tilde{w}_k^{(n)} \cdot \left( \sum_{i=1}^{N} \tilde{w}_k^{(i)} \right)^{-1}. \tag{6}$$

(3) *Resampling:* compute the effective sample size,

$$N_{\text{eff}} = \left( \sum_{n=1}^{N} \left( w_k^{(n)} \right)^2 \right)^{-1}. \tag{7}$$

If $N_{\text{eff}}$ is above some predefined threshold $N_{\text{thr}}$, simply define $\mathbf{X}_k^{(n)} = \tilde{\mathbf{X}}_k^{(n)}$ $\forall n$. Otherwise, draw $N$ new samples $\mathbf{X}_k^{(n)}$, $n \in \{1, \ldots, N\}$, from the existing set of particles $\{\tilde{\mathbf{X}}_k^{(i)}\}_{i=1}^{N}$ according to their weights $w_k^{(i)}$, then reset the weights to uniform values: $w_k^{(n)} = 1/N$ $\forall n$.

**Result:** the set $\{(\mathbf{X}_k^{(n)}, w_k^{(n)})\}_{n=1}^{N}$ is approximately distributed as the posterior density $p(\mathbf{X}_k \mid \mathbf{Y}_{1:k})$.

Algorithm 1: Generic bootstrap PF algorithm.

### 3.1. Target dynamics

In order to remain consistent with previous literature [2, 3], a Langevin process is used to model the target dynamics in (2a). This model is typically used to characterise various types of stochastic motion, and it has proved to be a good choice for acoustic speaker tracking. The source motion in each of the Cartesian coordinates is assumed to be an independent first-order process, which can be described by the following equation:

$$\mathbf{X}_k = \begin{bmatrix} 1 & 0 & aT_{\text{U}} & 0 \\ 0 & 1 & 0 & aT_{\text{U}} \\ 0 & 0 & a & 0 \\ 0 & 0 & 0 & a \end{bmatrix} \cdot \mathbf{X}_{k-1} + \begin{bmatrix} bT_{\text{U}} & 0 \\ 0 & bT_{\text{U}} \\ b & 0 \\ 0 & b \end{bmatrix} \cdot \mathbf{u}_k, \tag{8a}$$

with the noise variable

$$\mathbf{u}_k \sim \mathcal{N}\left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right), \tag{8b}$$

where $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the density of a multidimensional Gaussian random variable with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. The parameter $T_{\text{U}}$ corresponds to the time interval separating two consecutive updates of the particle

filter, and the other model parameters in (8) are defined as

$$a = \exp(-\beta T_{\mathrm{U}}), \qquad b = \overline{v}\sqrt{1-a^2}, \tag{9}$$

with $\overline{v}$ the steady-state velocity parameter and $\beta$ the rate constant.

### 3.2. Likelihood function[1]

Experimental results from previous research carried out on particle filtering for ASLT have shown that steered beamforming (SBF) delivers an improved tracking performance compared to TDE-based methods [2, 16]. Hence, the SBF principle is here also used as a basis for the derivation of the likelihood function. With $F_m(\omega) = \mathcal{F}\{f_m(t)\}$ the Fourier transform of the signal data from the $m$th sensor, and with $\|\cdot\|$ denoting the Euclidean norm, the output $\mathcal{P}(\boldsymbol{\ell})$ of a delay-and-sum beamformer steered to the location $\boldsymbol{\ell} = [x \;\; y]^{\mathrm{T}}$ is given as

$$\mathcal{P}(\boldsymbol{\ell}) = \int_{\Omega} \left| \sum_{m=1}^{M} W_m(\omega) F_m(\omega) \mathrm{e}^{j\omega\|\boldsymbol{\ell}-\boldsymbol{\ell}_m\|/c} \right|^2 \mathrm{d}\omega, \tag{10}$$

where $\boldsymbol{\ell}_m = [x_m \;\; y_m]^{\mathrm{T}}$ is the known position of the $m$th microphone, $W_m(\cdot)$ is a frequency weighting term, and $\Omega$ corresponds to the frequency range of interest, which is typically defined as $\Omega = \{\omega \mid 2\pi \cdot 300\,\mathrm{Hz} \leqslant \omega \leqslant 2\pi \cdot 3000\,\mathrm{Hz}\}$ for speech processing applications. In the following, the term $W_m(\cdot)$ is computed according to the phase transform (PHAT) weighting [17], for $m \in \{1, \ldots, M\}$,

$$W_m(\omega) = |F_m(\omega)|^{-1}. \tag{11}$$

For a given state $\mathbf{X}$, the likelihood function $p(\mathbf{Y} \mid \mathbf{X})$ measures the probability of receiving the data $\mathbf{Y}$. The SBF formula given in (10) effectively measures the level of acoustic energy that originates from a given focus location. The likelihood function should hence be chosen to reflect the fact that peaks in the SBF output $\mathcal{P}(\cdot)$ correspond to likely source locations, as well as the fact that, occasionally, there may be no peak in the SBF output corresponding to the true source due, for instance, to the effects of disturbances such as reverberation. The position of the peaks may also have slight errors due to noise or inaccurate sensor calibration. Based on these considerations, one approach to defining the likelihood function is to first select the positions $\widehat{\boldsymbol{\ell}}_\theta$, $\theta \in \{1, \ldots, \Theta\}$, of the $\Theta$ largest local maxima in the current SBF output. The generic observation variable $\mathbf{Y}$ is then typically defined as the set containing the selected SBF peak locations:

$$\mathbf{Y} \triangleq \{\widehat{\boldsymbol{\ell}}_1, \ldots, \widehat{\boldsymbol{\ell}}_\Theta\}, \tag{12}$$

and the following $\Theta + 1$ hypotheses can be considered:

$$\begin{aligned} &\mathcal{H}_\theta: \text{ SBF peak at location } \widehat{\boldsymbol{\ell}}_\theta \text{ is due to true source,} \\ &\mathcal{H}_0: \text{ no peak in the SBF output is due to true source,} \end{aligned} \tag{13}$$

with $\theta \in \{1, \ldots, \Theta\}$. The likelihood function is then given as follows:

$$p(\mathbf{Y} \mid \mathbf{X}) = \sum_{i=0}^{\Theta} q_i \cdot p(\mathbf{Y} \mid \mathbf{X}, \mathcal{H}_i), \tag{14}$$

with $q_i = p(\mathcal{H}_i \mid \mathbf{X})$, $i \in \{0, \ldots, \Theta\}$, the prior probabilities of the hypotheses. Without prior knowledge regarding the occurrence of each hypothesis, these probabilities are usually assumed equal and independent of the source location:

$$q_\theta = \frac{1 - q_0}{\Theta}, \quad \theta \in \{1, \ldots, \Theta\}. \tag{15}$$

Assuming statistical independence between different peak locations in the SBF measurement, the conditional terms on the right-hand side of (14) are given as follows:

$$p(\mathbf{Y} \mid \mathbf{X}, \mathcal{H}_i) = \prod_{\theta=1}^{\Theta} p(\widehat{\boldsymbol{\ell}}_\theta \mid \mathbf{X}, \mathcal{H}_i), \quad i \in \{0, \ldots, \Theta\}. \tag{16}$$

In a diffuse sound field comprising many different frequency components, such as the sound field resulting from reverberation, the energy density can be assumed uniform throughout the considered enclosure [18]. This means that given hypothesis $\mathcal{H}_0$, maximising the SBF output will result in a random location distributed uniformly across the state space. Given $\mathcal{H}_\theta$, $\theta \neq 0$, the likelihood of a measurement originating from the source is typically modeled as a Gaussian PDF with variance $\sigma_{\mathbf{Y}}^2$, to account for measurement and calibration errors. Thus, with $\mathcal{N}(\boldsymbol{\xi}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denoting a Gaussian density with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ evaluated at $\boldsymbol{\xi}$, the likelihood for each SBF peak can be defined as follows:

$$p(\widehat{\boldsymbol{\ell}}_\theta \mid \mathbf{X}, \mathcal{H}_i) = \begin{cases} \mathcal{N}(\boldsymbol{\ell}_{\mathbf{X}}; \widehat{\boldsymbol{\ell}}_\theta, \sigma_{\mathbf{Y}}^2 \mathbf{I}) & \text{if } \theta = i, \\ \mathcal{U}_{\mathcal{D}}(\boldsymbol{\ell}_{\mathbf{X}}) & \text{otherwise,} \end{cases} \tag{17}$$

where $\boldsymbol{\ell}_{\mathbf{X}} = [x \;\; y]^{\mathrm{T}}$ corresponds to the top half of the state vector $\mathbf{X}$, $\mathbf{I}$ is the $2 \times 2$ identity matrix, and with $\mathcal{U}_{\mathcal{D}}(\cdot)$ the uniform PDF over the considered enclosure domain $\mathcal{D} = \{(x, y) \mid x_{\min} \leqslant x \leqslant x_{\max},\ y_{\min} \leqslant y \leqslant y_{\max}\}$.

The derivations presented so far suffer from a major drawback: the SBF output has to be computed across the entire domain $\mathcal{D}$ in order to find $\Theta$ local maxima $\widehat{\boldsymbol{\ell}}_\theta$, which leads to a considerable computational load in practical implementations. One approach that circumvents this drawback is based on the concept of a "pseudo-likelihood," as introduced previously in [2]. This concept relies on the idea that the SBF output $\mathcal{P}(\cdot)$ itself can be used as a measure of likelihood. Adopting this approach implicitly reduces the number of hypotheses to the following two events:

$$\begin{aligned} &\mathcal{H}_0: \text{ SBF measurement originates from clutter,} \\ &\mathcal{H}_1: \text{ SBF measurement originates from true source,} \end{aligned} \tag{18}$$

[1] For clarity, the frame subindex $k$ is omitted in this section, implicitly assuming that all variables of interest refer to the current frame of data $k$.

with respective prior probabilities $q_0 = p(\mathcal{H}_0 \mid \mathbf{X})$ and $q_1 = p(\mathcal{H}_1 \mid \mathbf{X}) = 1 - q_0$. Note also that the pseudo-likelihood approach implicitly redefines the observation variable $\mathbf{Y}$ as the SBF output function $\mathcal{P}(\cdot)$ itself; $\mathbf{Y}$ hence does not correspond to a set of SBF peaks as given in (12) anymore. On the basis of (14), (16) and (17), the new likelihood function can be derived as

$$p(\mathbf{Y} \mid \mathbf{X}) = q_0 \cdot \mathcal{U}_{\mathcal{D}}(\boldsymbol{\ell}_{\mathbf{X}}) + \gamma(1 - q_0) \cdot (\mathcal{P}(\boldsymbol{\ell}_{\mathbf{X}}))^r, \tag{19}$$

where the nonlinear exponent $r$ is used to help shape the SBF output to make it more amenable to source tracking [2].[2] The parameter $\gamma$ in (19) is a normalisation constant ensuring that $\mathcal{P}(\cdot)$ is suitable for a use as density function, and computed in theory such that

$$\gamma \cdot \iint_{\mathcal{D}} (\mathcal{P}(\boldsymbol{\ell}))^r \, \mathrm{d}\boldsymbol{\ell} = 1. \tag{20}$$

However, the computation of $\gamma$ according to (20) here again involves the computation of $\mathcal{P}(\cdot)$ across the entire domain $\mathcal{D}$, which is not desirable. In [2], this issue was solved by defining $q_0 = 0$ and $\gamma = 1$, arguing that the SBF measurements are always positive and that the update step of the PF algorithm would ensure that the particle weights are suitably normalised. In the present work however, a proper normalisation parameter $\gamma$ in the pseudo-likelihood defined by (19) is necessary, since $q_0 \neq 0$ will be assumed in the following developments. Consequently, we propose a normalisation coefficient based on a different principle. As derived previously, a Gaussian likelihood model would typically first determine the global maximum $\widehat{\boldsymbol{\ell}}$ of $\mathcal{P}(\cdot)$, and subsequently define $p(\mathbf{Y} \mid \mathbf{X})$ as a Gaussian density centered on $\widehat{\boldsymbol{\ell}}$ and with a certain variance $\sigma_{\mathbf{Y}}^2$, see (17). For the pseudo-likelihood approach, we hence propose to normalise $\mathcal{P}(\cdot)$ so that its maximum value is equal to the peak value of this Gaussian PDF:

$$\gamma \cdot \max_{\boldsymbol{\ell} \in \mathcal{D}} \{(\mathcal{P}(\boldsymbol{\ell}))^r\} = \max_{\boldsymbol{\ell} \in \mathcal{D}} \{\mathcal{N}(\boldsymbol{\ell}; \widehat{\boldsymbol{\ell}}, \sigma_{\mathbf{Y}}^2 \mathbf{I})\} = (2\pi\sigma_{\mathbf{Y}}^2)^{-1}. \tag{21}$$

The value of the parameter $\gamma$ can be derived from (21) as follows. Due to the PHAT weighting in (11), and using the representation $F_m(\omega) = |F_m(\omega)| \cdot \mathrm{e}^{j\phi_m(\omega)}$, the SBF output computed according to (10) becomes

$$\mathcal{P}(\boldsymbol{\ell}) = \int_{\Omega} \left| \sum_{m=1}^{M} \mathrm{e}^{j\Phi_m(\omega)} \right|^2 \mathrm{d}\omega, \tag{22}$$

with $\Phi_m(\omega) = \phi_m(\omega) + \omega\|\boldsymbol{\ell} - \boldsymbol{\ell}_m\|c^{-1}$. According to the Cauchy-Schwarz inequality, the SBF output values are thus bounded as follows:

$$\begin{aligned} \mathcal{P}(\boldsymbol{\ell}) &\leqslant \int_{\Omega} \left( \sum_{m=1}^{M} \left| \mathrm{e}^{j\Phi_m(\omega)} \right| \right)^2 \mathrm{d}\omega \\ &= M^2(\omega_{\max} - \omega_{\min}), \end{aligned} \tag{23}$$

where $\omega_{\max}$ and $\omega_{\min}$ are the upper and lower limits of the frequency range $\Omega$, respectively. Using the result of (23), the normalisation constant in (21) finally becomes

$$\gamma = \frac{1}{2\pi\sigma_{\mathbf{Y}}^2 M^{2r} (\omega_{\max} - \omega_{\min})^r}. \tag{24}$$

The normalisation process described here ensures that the two PDFs in the mixture likelihood definition of (19) are properly scaled with respect to each other.

### 3.3. PF algorithm outputs

For each frame $k$ of input data, the particle filter delivers the following two outputs. First, an estimate $\widehat{\boldsymbol{\ell}}_{\mathbf{X},k}$ of the source position is computed according to (5b):

$$\widehat{\boldsymbol{\ell}}_{\mathbf{X},k} = \sum_{n=1}^{N} w_k^{(n)} \boldsymbol{\ell}_{\mathbf{X},k}^{(n)}, \tag{25}$$

where $\boldsymbol{\ell}_{\mathbf{X},k}^{(n)} = [x_k^{(n)} \;\; y_k^{(n)}]^{\mathrm{T}}$ corresponds to the location information in the $n$th particle vector. The second output is a measure of the confidence level in the PF estimates, which can be obtained by computing the standard deviation of the particle set:

$$\varsigma_k = \sqrt{\sum_{n=1}^{N} w_k^{(n)} \|\boldsymbol{\ell}_{\mathbf{X},k}^{(n)} - \widehat{\boldsymbol{\ell}}_{\mathbf{X},k}\|^2}. \tag{26}$$

The parameter $\varsigma_k$ provides a direct assessment of how reliable the PF considers its current source position estimate to be.

## 4. VOICE ACTIVITY DETECTION

The voice activity detector (VAD) employed here relies on an estimate of the instantaneous signal-to-noise ratio (SNR) in the current block of data [12]. It assumes that the data recorded at the microphones is a combination of the speech signal and noise:

$$f_m(t) \triangleq s_m(t) + v_m(t), \quad m \in \{1, \ldots, M\}, \tag{27}$$

where the signal $s_m(\cdot)$ and noise $v_m(\cdot)$ are uncorrelated. It is further assumed that the microphone signals are bandlimited and sampled in time.

The scheme works on the basis of the expected noise power spectral density, which is estimated during nonspeech periods. The estimated noise level is then used during periods of speech activity to estimate the SNR from the observed signal. The assumption is that the speaker is active when the signal level is sufficiently higher than the noise level: the speech versus nonspeech decision is made by comparing the mean SNR to a threshold, where the SNR average is taken over the considered frequency domain. The spectral resolution is defined to be lower than the frame length in order to decrease the variance of the signal power estimates. The specific application considered in this work makes it possible to reduce the variance further by averaging over multiple microphones. The frame length $L$ is chosen such that the propagation delay to the different microphones does not impact significantly on the power estimate.

[2] Using $r > 1$ typically increases the sharpness of the peaks while reducing the background noise variance in the SBF measurements.

### 4.1. SNR estimation

The instantaneous, reduced-resolution estimate $P_{f,d}(k)$ of the power spectral density for the $d$th frequency band and the $k$th frame of data from the microphones is obtained according to

$$P_{f,d}(k) = \frac{1}{M}\sum_{m=1}^{M}\int_{\Omega_d}\varphi(\omega)\left|\frac{1}{L}\sum_{l=kL-L+1}^{kL} f_m(l)\mathrm{e}^{jl\omega}\right|^2 \mathrm{d}\omega, \tag{28}$$

where the window function $\varphi(\omega)$ is here chosen to deemphasise the lower frequency range, in order to suppress frequencies with high noise content. The integration regions $\Omega_d$, $d \in \{1,\ldots,D\}$, divide the frequency space into a small number (typically eight) of nonoverlapping bands of equal width. The background noise power $P_{v,d}$ is assumed to vary slowly in relation to the speech power. In practice, a time-varying estimate $\widehat{P}_{v,d}(k)$ of $P_{v,d}$ is obtained by averaging $P_{f,d}(\cdot)$ over time during the nonspeech periods detected by the algorithm. An initial estimate of $P_{v,d}$ is typically obtained during a short algorithm initialisation phase, carried out during a period of background noise only.

The instantaneous SNR for frequency band $d$ is calculated according to

$$\psi_d(k) = \frac{P_{f,d}(k)}{P_{v,d}} - 1. \tag{29}$$

During nonspeech periods, we have $P_{f,d}(k) \approx P_{v,d}$, and the variance of the instantaneous SNR becomes

$$\sigma^2_{v,d} = \mathbb{E}\left\{\left(\psi_d(k) - \mathbb{E}\{\psi_d(k)\}\right)^2\right\} = \mathbb{E}\{\psi_d^2(k)\}, \tag{30}$$

where $\mathbb{E}\{\cdot\}$ represents the statistical expectation. Thus, an estimate $\widehat{\sigma}^2_{v,d}(k)$ of the background noise variance can be found by averaging the square of the instantaneous SNR during nonspeech periods.

### 4.2. Statistical detection

The speaker is assumed to be active during the $k$th frame when the instantaneous SNR $\psi_d(k)$ is higher than a threshold $\eta_d$. The threshold can be derived by considering the problem as a hypothesis test:

$$\begin{aligned}\mathcal{H}_0 &: \psi_d(k) = \frac{P_{v,d}(k)}{P_{v,d}} - 1,\\ \mathcal{H}_1 &: \psi_d(k) = \frac{P_{v,d}(k) + P_{s,d}(k)}{P_{v,d}} - 1 = \frac{P_{f,d}(k)}{P_{v,d}} - 1,\end{aligned} \tag{31}$$

where $P_{s,d}(k)$ and $P_{v,d}(k)$ are the instantaneous speech signal and noise power, respectively, the null hypothesis $\mathcal{H}_0$ denotes nonspeech, and $\mathcal{H}_1$ the alternative.

The PDF for the instantaneous SNR estimates during nonspeech can be defined as

$$p(\psi_d(k) \mid \mathcal{H}_0) = \frac{1}{\sqrt{2\pi\sigma^2_{v,d}}}\exp\left(\frac{-\psi_d^2(k)}{2\sigma^2_{v,d}}\right), \tag{32}$$

assuming that the estimates are Gaussian distributed. This assumption is not always correct, but works well as an approximation under real conditions [12]. From (32), the probability of false alarm $P_{\mathrm{FA}}$, that is, speech reported during nonspeech period, can then be formulated as

$$P_{\mathrm{FA}} = \Pr\{\eta_d < \psi_d(k) \mid \mathcal{H}_0\} \tag{33a}$$

$$= \int_{\eta_d}^{\infty}\frac{1}{\sqrt{2\pi\sigma^2_{v,d}}}\exp\left(\frac{-\psi_d^2(k)}{2\sigma^2_{v,d}}\right)\mathrm{d}\psi_d(k). \tag{33b}$$

By rearranging (33b) and solving for $\eta_d$ we obtain

$$\eta_d = \sqrt{2\sigma^2_{v,d}}\cdot \mathrm{erfc}^{-1}(2P_{\mathrm{FA}}), \tag{34}$$

where $\mathrm{erfc}(\cdot)$ is the complementary error function [19]. In a practical implementation, a time-varying estimate $\widehat{\eta}_d(k)$ of the threshold is obtained by using the estimated background noise variance $\widehat{\sigma}^2_{v,d}(k)$. Finally, the binary VAD decision $\rho(k)$ for speech is made by comparing the mean instantaneous SNR to the mean threshold, where the average is taken over all frequency bands:

$$\rho(k) = \begin{cases} 1 & \text{if } \sum_{d=1}^{D}\psi_d(k) > \sum_{d=1}^{D}\eta_d(k),\\ 0 & \text{otherwise,}\end{cases} \tag{35}$$

where 1 denotes speech and 0 nonspeech.

Note that the operation of the algorithm depends on the state of its own output for determining when to start estimating the background noise power. During the SNR estimation process, a hangover scheme based on a state machine is therefore used in order to reduce the probability of speech entering the background noise estimate [12]. However, if the background noise power changes rapidly, the algorithm may enter a state where it will provide erroneous decisions, which is a limitation inherent to the considered VAD method. Experimental tests have however shown that this happens very rarely in practice, and that the algorithm is able to recover by itself in such cases after a short transitional period.

## 5. FUSION OF VAD MEASUREMENTS

A straightforward approach to merging different measurement modalities within the PF framework is via the definition of a combined likelihood function. This representation however would fuse both the VAD and SBF measurements at the same algorithmic level, implicitly assuming statistical independence between these two types of observation. In the context of the specific ASLT problem considered in this work, this is not completely justified: intuitively, if the VAD classifies the current frame of data as nonspeech, the corresponding SBF measurement is likely to be unreliable in terms of source localisation accuracy. We hence adopt a different approach to the fusion problem, as described in the following.

The output of the VAD can be linked to the probability of the hypotheses in (18) in an obvious manner. For instance, considered as an indication of the likelihood that the current

SBF observation originates from clutter only, the variable $q_0$ explicitly measures the probability of the acoustic source being inactive. Likewise, $q_1 = 1 - q_0$ corresponds to the likelihood of the source being active, an estimate of which is delivered by the VAD. Therefore, instead of setting the variable $q_0$ to a constant value in the design of the algorithm as done in [2, 3], we propose to use a time-varying $q_0$ parameter based on the output of the VAD as follows:

$$q_0(k) = 1 - \alpha(k), \tag{36}$$

where $\alpha(k) \in [0, 1]$ is derived from the state of the VAD algorithm. The generic algorithm resulting from (36) and from the developments in Section 3 will be denoted PF-VAD from here on.

Three different methods for deriving the parameter $\alpha(k)$ form the VAD algorithm are suggested. These are defined as follows:

$$\begin{aligned} \alpha_{\text{SNR}}(k) &= \frac{2}{\pi} \arctan\left(\overline{\psi}(k)\right), \\ \alpha_{\text{SP}}(k) &= \frac{\overline{P}_v(k) \cdot \overline{\psi}(k)}{\max\limits_{i<k} \left(\alpha_{\text{SP}}(i)\right)}, \\ \alpha_{\text{BIN}}(k) &= \rho(k), \end{aligned} \tag{37}$$

with the following definitions:

$$\begin{aligned} \overline{\psi}(k) &= \sqrt{\frac{1}{D} \sum_{d=1}^{D} \psi_d(k)}, \\ \overline{P}_v(k) &= \sqrt{\frac{1}{D} \sum_{d=1}^{D} \hat{P}_{v,d}(k)}. \end{aligned} \tag{38}$$

The first method, that is, the VAD output $\alpha_{\text{SNR}}(\cdot)$, maps the mean instantaneous SNR gain level (a number between 0 and $\infty$) to $\alpha(\cdot)$ through bilinear transformation. The reasoning behind this approach is that a high SNR should indicate that the signal received at the microphones contains information useful to the tracking algorithm. The second method, $\alpha_{\text{SP}}(\cdot)$, calculates an estimate of the speech signal level. The normalisation with respect to all previous maximum signal levels is carried out in order to remove the influence of the absolute signal level at the microphones. This approach effectively discards the noise level information and assumes that only the speech signal level information is useful to the tracking algorithm. The last method, $\alpha_{\text{BIN}}(\cdot)$, simply uses the binary output $\rho(\cdot)$ from the VAD as $\alpha(\cdot)$. The "all-or-nothing" approach used by this method potentially discards a substantial amount of useful information. It however still represents an alternative of potential interest, and is included here for the purpose of providing a performance comparison baseline.

Figure 1 shows an example of the different VAD outputs defined above. The curves obtained with these VAD methods will typically differ from each other as a function of the specific noise and reverberation level contained in the input signals. Compared to the binary output $\alpha_{\text{BIN}}(\cdot)$, the use of soft VAD information with $\alpha_{\text{SNR}}(\cdot)$ and $\alpha_{\text{SP}}(\cdot)$ allows the PF to track the source in a more subtle manner. For instance, a VAD output value $0 < \alpha(\cdot) < 1$ effectively indicates that the input signals may be partly corrupted by disturbance sources, and that the current SBF observation might not be fully accurate. The PF can then take account of this fact and use more caution when updating the particle set, and hence, when determining the source location estimate. With the binary VAD output $\alpha_{\text{BIN}}(\cdot)$, the source tracking process is basically turned fully on or off based on $\rho(\cdot)$ (hard decisions), which may not be advantageous when a high level of noise and/or reverberation is present. In the next section, results from experimental simulations of the PF-VAD method will determine which one of these three approaches delivers the best tracking performance.

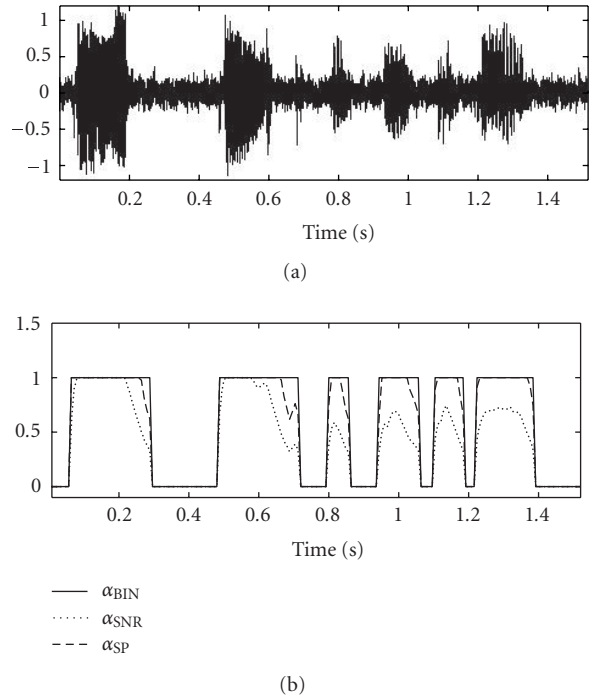


Figure 1: Practical example of three considered VAD methods. (a) Input signal data. (b) Resulting VAD outputs.

## 6. EXPERIMENTAL RESULTS

This section presents some examples of the tracking results obtained with the proposed PF-VAD algorithm. The various parameters of the PF-VAD implementation were optimised empirically and set to the following values: the number of particles was set to $N = 50$, the effective sample size threshold $N_{\text{thr}} = 37.5$, the standard deviation of the observation density was defined as $\sigma_{\Upsilon} = 0.15$ m, and the nonlinear exponent was set to $r = 2$. Following standard definitions (see, e.g., [2, 3]), the PF-VAD implementation made use of the propagation model parameters $\overline{v} = 0.8$ m/s and $\beta = 10$ Hz. The VAD parameters were defined as $P_{\text{FA}} = 0.03$ and $D = 8$. The audio signals were sampled with a frequency of 16 kHz and processed in nonoverlapping frames of $L = 256$ samples each.

For comparison purposes, the performance assessment given in this section also includes results from the SBF-PL algorithm, a sound source tracking scheme previously proposed in [2]. The SBF-PL method relies on a particle filtering approach similar to that presented in this work, but does not include any VAD measurements. The reader is referred to [2] for a more detailed description of the SBF-PL implementation, and to [16] for a summary of its practical performance results and a comparison with other tracking methods.

### 6.1. Assessment parameters

The experimental results make use of the following parameters to assess the tracking accuracy of the considered methods. The PF estimation error for the current frame is

$$\varepsilon_k = ||\boldsymbol{\ell}_{S,k} - \widehat{\boldsymbol{\ell}}_{\mathbf{X},k}||, \tag{39}$$

where $\boldsymbol{\ell}_{S,k}$ is the ground-truth source position at time $k$. In order to assess the overall performance of the developed algorithm over a given sample of audio data, the average error is simply computed as

$$\overline{\varepsilon} = \frac{1}{K}\sum_{k=1}^{K}\varepsilon_k, \tag{40}$$

with $K$ representing the total number of frames in the considered audio sample. The standard deviation parameter $\varsigma_k$, see (26), is also used here as an overall indication of the PF tracking performance in the following results presentation.

### 6.2. Image method simulations

The proposed PF algorithm was put to the test using synthetic reverberant audio data generated using the image source method [20]. The results presented in this section were obtained using audio data generated with the source trajectory, source signal, and microphone setup depicted in Figure 2. The dimension of the enclosure was set to 3 m × 3 m × 2.5 m, and the height of the microphones, as well as that of the source, was defined as 1.5 m.

Figure 3 presents some typical results obtained with the two considered ASLT methods (where PF-VAD uses the speech-based VAD output $\alpha_{SP}$), using the setup of Figure 2 with a reverberation time $T_{60} \approx 0.1$ s and input SNR of approximately 15 dB. This figure clearly illustrates the most significant outcome of the PF-VAD implementation. Fusing the VAD measurements within the PF framework effectively allows the tracking algorithm to put more emphasis on the considered dynamics model in (8) when spreading the particles during nonspeech periods, while at the same time reducing the importance of the SBF observations due to the fact that no useful information can be derived from them when the speaker is inactive. This consequently allows the PF to keep track of the silent target, and to resume tracking successfully when the speaker becomes active again. This can be distinctly noticed with the consistent increase of the $\varsigma_k$ values for PF-VAD (Figure 3(b)) during significant gaps in the speech signal. This specific effect originates from the influence of the VAD measurements on the effective sample size parameter $N_{\text{eff}}$. Figure 4(b) shows an example of the $N_{\text{eff}}$ values computed during one run of PF-VAD versus time. As described in step 3 of Algorithm 1, the parameter $N_{\text{eff}}$ is reset to $N$ after the resampling stage is carried out, and the result in Figure 4 thus provides an overall view of the resampling frequency. This plot demonstrates how the VAD output "freezes" the $N_{\text{eff}}$ value during nonspeech periods, effectively decreasing the occurrence of the particle resampling step, which in turn leads to a spatial evolution of the particles according to the dynamics model only.

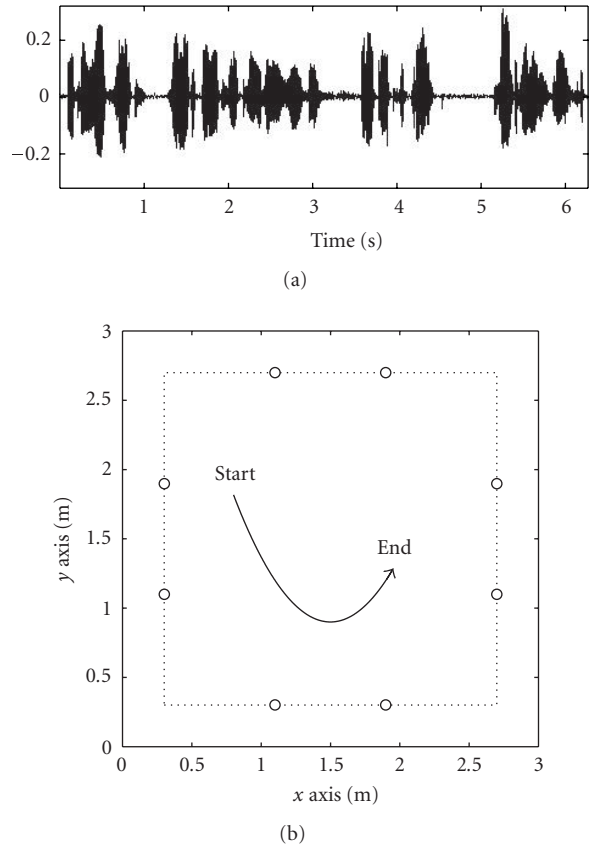


Figure 2: Setup for image method simulations. (a) Source signal. (b) Microphone positions (∘) and parabolic source trajectory.

As an important consequence of this fact, the standard deviation $\varsigma_k$ delivered by PF-VAD effectively reflects a "true" confidence level, that is, in keeping with the estimation accuracy, and can be hence directly used as an indication of the reliability of the PF estimates. For instance, an obvious add-on to the PF-VAD method would be to simply discard the PF location estimates whenever $\varsigma_k$ is above a predefined threshold.

On the other hand, the more or less constant resampling frequency implemented as part of the SBF-PL method precludes this desired behaviour, meaning that the particles always remain very concentrated spatially. This essentially implies that during nonspeech periods, the SBF-PL particle filter continues its tracking as if the speaker was still active, and

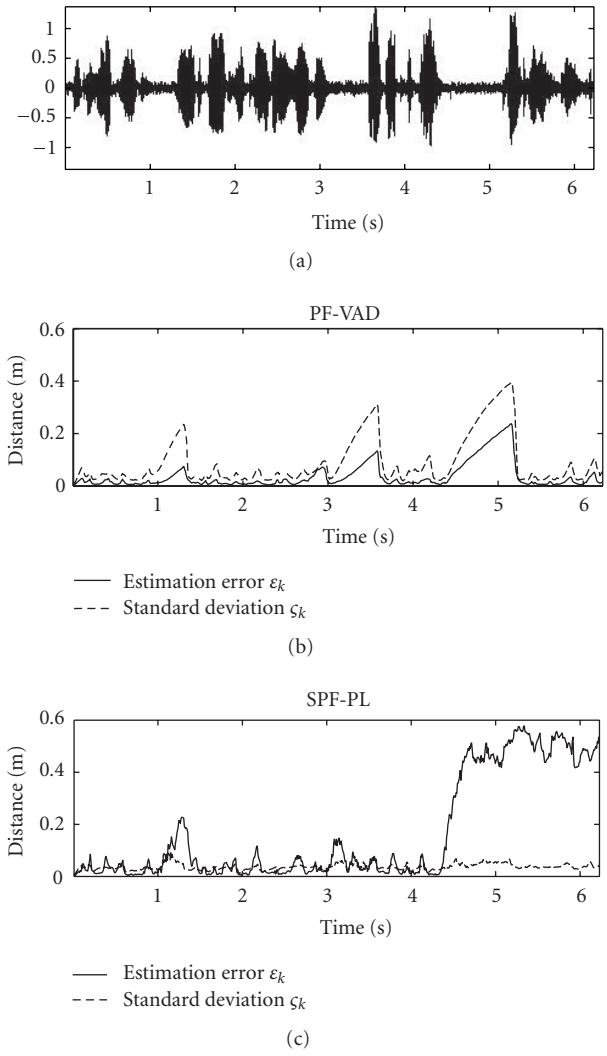


Figure 3: Tracking result examples for two ASLT methods, for $T_{60} \approx 0.1$ s and SNR $\approx 15$ dB. (a) Example of microphone signal. (b) and (c) Estimation error and standard deviation for PF-VAD and SBF-PL (results averaged over 100 simulation runs).

is hence much more likely to be driven off-track by the effects of reverberation and additive noise. An example of such a scenario is shown in Figure 3(c), where SBF-PL loses track of the speaker at the end of the simulation due to a significant gap in the speech signal.

Figures 5 and 6 present the average tracking results obtained for the proposed PF-VAD algorithm, as well as a comparison with the previously developed SBF-PL method. These plots show the average error $\overline{\varepsilon}$ computed over a range of input SNR values (Figure 5) and reverberation times (Figure 6). Different $T_{60}$ values were achieved by appropriately setting the walls' reflection coefficients in the image method implementation. Statistical averaging was performed due to the random nature of the PF implementation, and the results depicted in these figures represent the average over 100 simulation runs of the considered algorithms, using the above-mentioned image method setup.

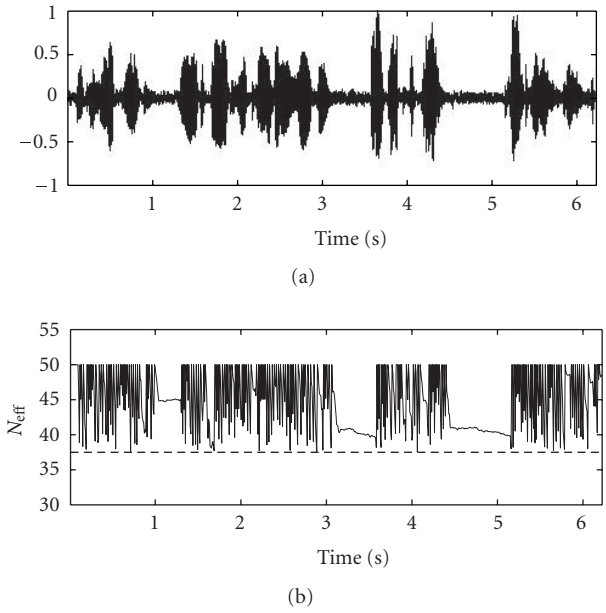


Figure 4: Overview of the resampling frequency during one run of PF-VAD. (a) Example of input signal used for this simulation, and (b) effective sample size parameter $N_{\text{eff}}$ versus time (dashed line: threshold $N_{\text{thr}}$).

These results clearly demonstrate the superiority of the proposed PF-VAD algorithm. The SBF-PL method consistently exhibits a larger average error due to track losses occurring as a result of significant gaps in the considered speech signal (see the source signal plotted in Figure 2(a)), which the PF-VAD implementation manages to avoid. Also, it must be kept in mind that the PF-VAD results shown in Figures 5 and 6 correspond to the mean error $\overline{\varepsilon}$ computed over the entire length of the considered audio sample. This typically also includes periods where the PF has a low confidence level in its estimates. As mentioned earlier, the average performance of PF-VAD would improve even further if tracking estimates were discarded when $\varsigma_k$ is above a predefined threshold.

In regards to a comparison of the three tested VAD schemes with each other, it can be seen from Figures 5 and 6 that the speech-based VAD scheme $\alpha_{\text{SP}}$ generally tends to yield the best overall tracking performance, given the specific test setup considered in this section. This result suggests that the most useful information from a tracking point of view relies more on the amount of speech present during a given time frame, rather than the speech-to-noise ratio, which, for instance, may become large despite a small speech signal level in some circumstances.

### 6.3. Real-time implementation and real audio tracking

While the image method simulations presented in the previous section are useful to gauge the proposed algorithm's ability to deal with the considered ASLT problem, only a real-time implementation, used in conjunction with real audio signals, is able to provide a full insight into how suitable the

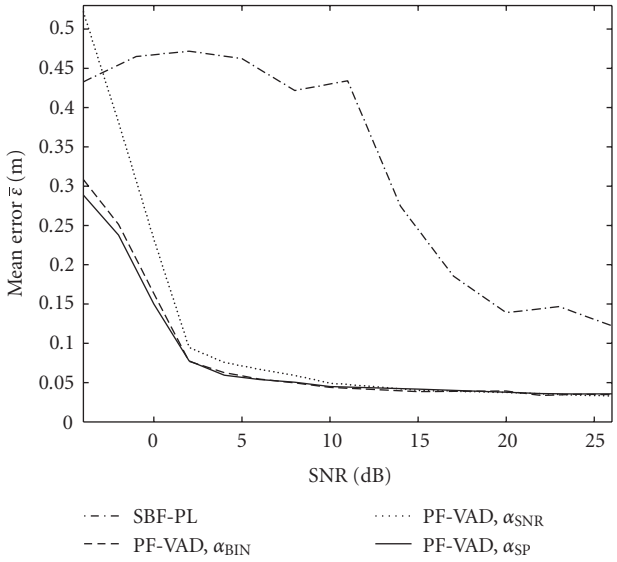


FIGURE 5: Average tracking error versus input signal SNR, for $T_{60} \approx 0.1$ s (results averaged over 100 simulation runs).

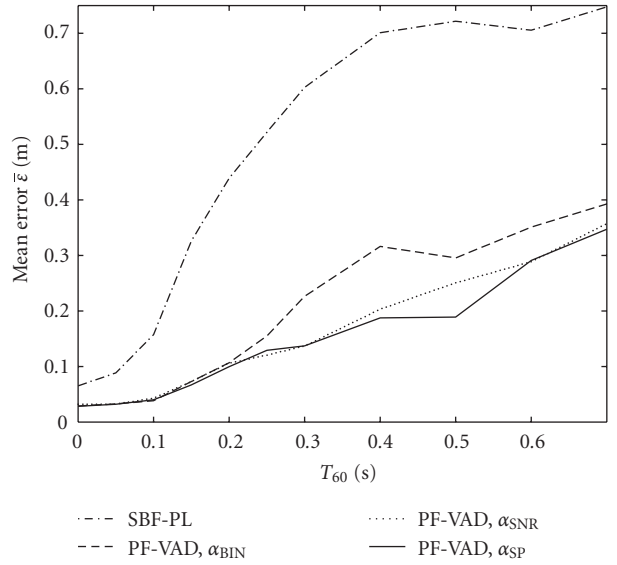


FIGURE 6: Average tracking error versus reverberation time $T_{60}$, with input SNR of about 20 dB (results averaged over 100 simulation runs).

algorithm is for practical applications. Such an implementation has also been carried out in the frame of this research. However, for the sake of conciseness, details of this implementation and of the real audio tracking results are presented elsewhere, and only a brief review of these results is presented here.

The PF-VAD algorithm was implemented on a standard 1.8 GHz IBM-PC running under Linux, used in conjunction with an array of eight microphones sampled at 16 kHz. An analysis of the algorithm showed that an implementation with 100 particles results in a computational complexity of 71.5 M floating-point operations per second (FLOPS), resulting in a CPU load during execution of about 5%. These results hence demonstrate the suitability of the PF-VAD method for real-time processing on low-power embedded systems using all-purpose hardware and software. Full details of this real-time implementation can be found in [21].

A full tracking performance assessment of the PF-VAD algorithm was also conducted using samples of real audio data, recorded in a reverberant environment. A microphone array, similar to that shown in Figure 2, was set up in a room with dimensions 3.5 m × 3.1 m × 2.2 m and a practical reverberation time of $T_{60} \approx 0.3$ s (frequency-averaged up to 24 kHz). The experimental results using this practical setup are reported in [22], and confirm the improved efficiency of PF-VAD compared to SBF-PL when used in real-world circumstances.

## 7. CONCLUSION AND FUTURE WORK

This work is concerned with the problem of tracking a human speaker in reverberant and noisy environments by means of an array of acoustic sensors. We derived a PF-based method that integrates VAD measurements at a low level in the statistical algorithm framework. Provided the dynamics of the considered acoustic source are properly modeled, the proposed PF-VAD method greatly reduces the likelihood of a complete track loss during long silence gaps in the speech signal. The proposed algorithm hence provides an improved tracking performance for real-world implementations compared to previously derived PF methods. As a further result of the proposed implementation, the standard deviation of the particle set can now be used as a reliable indication of the filter's own estimation accuracy. The obvious limitation inherent to the current developments is that only one single speaker can be tracked at a time. This work will however serve as a basis for further research on the problem of multiple speaker tracking using the principle of microphone array beamforming.

## ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable suggestions and comments, as well as Alan Davis for the help provided in regards to the VAD scheme used in this paper. This work was supported by National ICT Australia (NICTA) and the Australian Research Council (ARC) under Grant no. DP0451111. NICTA is funded by the Australian Government's Department of Communications, Information Technology and the Arts, the Australian Research Council through Backing Australia's Ability, and the ICT Centre of Excellence programs.

## REFERENCES

[1] S. Gannot and T. G. Dvorkind, "Microphone array speaker localizers using spatial-temporal information," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 59625, 17 pages, 2006.

[2] D. B. Ward, E. A. Lehmann, and R. C. Williamson, "Particle filtering algorithms for tracking an acoustic source in a reverberant environment," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 6, pp. 826–836, 2003.
[3] J. Vermaak and A. Blake, "Nonlinear filtering for speaker tracking in noisy and reverberant environments," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '01)*, vol. 5, pp. 3021–3024, Salt Lake City, Utah, USA, May 2001.
[4] I. Potamitis, H. Chen, and G. Tremoulis, "Tracking of multiple moving speakers with multiple microphone arrays," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 520–529, 2004.
[5] T. G. Dvorkind and S. Gannot, "Speaker localization exploiting spatial-temporal information," in *Proceedings of the International Workshop on Acoustic Echo and Noise Control (IWAENC '03)*, pp. 295–298, Kyoto, Japan, September 2003.
[6] D. Bechler, M. Grimm, and K. Kroschel, "Speaker tracking with a microphone array using Kalman filtering," *Advances in Radio Science*, vol. 1, pp. 113–117, 2003.
[7] J. Chen, L. Shue, and W. Ser, "A new approach for speaker tracking in reverberant environment," *Signal Processing*, vol. 82, no. 7, pp. 1023–1028, 2002.
[8] Y. Huang, J. Benesty, and G. W. Elko, "Passive acoustic source localization for video camera steering," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '00)*, vol. 2, pp. 909–912, Istanbul, Turkey, June 2000.
[9] S. Doclo and M. Moonen, "Robust adaptive time delay estimation for speaker localization in noisy and reverberant acoustic environments," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1110–1124, 2003.
[10] X. Sheng and Y. H. Hu, "Sequential acoustic energy based source localization using particle filter in a distributed sensor network," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '04)*, vol. 3, pp. 972–975, Montreal, Québec, Canada, May 2004.
[11] J. C. Chen, K. Yao, and R. E. Hudson, "Acoustic source localization and beamforming: theory and practice," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 4, pp. 359–370, 2003.
[12] A. Davis, S. Nordholm, and R. Togneri, "Statistical voice activity detection using low-variance spectrum estimation and an adaptive threshold," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 2, pp. 412–424, 2006.
[13] B. Anderson and J. Moore, *Optimal Filtering*, Dover, New York, NY, USA, 2005.
[14] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
[15] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings, F: Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.
[16] E. A. Lehmann, D. B. Ward, and R. C. Williamson, "Experimental comparison of particle filtering algorithms for acoustic source localization in a reverberant room," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, vol. 5, pp. 177–180, Hong Kong, April 2003.
[17] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.
[18] R. Waterhouse, "Statistical properties of reverberant sound fields," *Journal of the Acoustical Society of America*, vol. 43, no. 6, pp. 1436–1444, 1968.
[19] S. Haykin, *Communication Systems*, John Wiley & Sons, New York, NY, USA, 3rd edition, 1994.
[20] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.
[21] A. M. Johansson, E. A. Lehmann, and S. Nordholm, "Real-time implementation of a particle filter with integrated voice activity detector for acoustic speaker tracking," in *Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems (APCCAS '06)*, Singapore, December 2006.
[22] E. A. Lehmann and A. M. Johansson, "Experimental performance assessment of a particle filter with voice activity data fusion for acoustic speaker tracking," in *Proceedings of the 7th IEEE Nordic Signal Processing Symposium (NORSIG '06)*, Reykjavik, Iceland, June 2006.

**Eric A. Lehmann** graduated in 1999 from the Swiss Federal Institute of Technology in Zurich (ETHZ), Switzerland, with a Diploma in electrical engineering (Master equivalent). He received the M.Phil. and Ph.D. degrees, both in electrical engineering, from the Australian National University (Canberra) in 2000 and 2004, respectively. After working as a Research Engineer for National ICT Australia (NICTA) in Canberra, he now holds a research position with the Western Australian Telecommunications Research Institute (WATRI) in Perth, Australia. His current scientific interests include acoustics, signal and speech processing, microphone arrays, and Bayesian estimation and tracking, with particular emphasis on the application of sequential Monte Carlo methods (particle filters).

**Anders M. Johansson** was born on February 10, 1974, in Sweden. He studied Telecommunications and Signal Processing at the Blekinge Technical University and received a Master's degree in electrical engineering in 2000. He held the position of Research Engineer at the Australian Telecommunications Research Institute from 2000 to 2002, and at the West Australian Telecommunications Research Institute, from 2002 to present, developing real-time software for research in the field of acoustic signal processing. His main fields of interest include acoustic source localisation, blind signal separation, real-time signal processing, and acoustics.