COMPARISON OF SRP-PHAT AND MULTIBAND-POPI ALGORITHMS FOR SPEAKER LOCALIZATION USING PARTICLE FILTERS

Tania Habib and Harald Romsdorfer[*]

Signal Processing and Speech Communication Lab
Graz University of Technology
Graz, Austria
{tania.habib, romsdorfer}@tugraz.at

ABSTRACT

The task of localizing single and multiple concurrent speakers in a reverberant environment with background noise poses several problems. One of the major problems is the severe corruption of the frame-wise localization estimates. To improve the overall localization accuracy, we propose a particle filter based tracking algorithm using the recently proposed Multiband Joint Position-Pitch (M-PoPi) localization algorithm as a frame wise likelihood estimate. To prove the performance of our approach, we tested it on real-world recordings of seven different speakers and of up to three concurrent speakers. We compared our new approach to the well-known SRP-PHAT algorithm as frame-wise likelihood estimates. Finally, we compared both particle filter based tracking algorithms with their frame-wise localization algorithms. The M-PoPi based particle filter tracking algorithm outperforms the SRP-PHAT based particle filter tracking algorithm. The comparison with their frame wise localization algorithms shows that this improved performance stems from the more robust M-PoPi frame wise localization estimate.

1. INTRODUCTION

Acoustic source localization using measurements from a microphone array has been an active area of research in recent years finding its applications in robotics, teleconferences, surveillance and object tracking.

There exist various approaches to localize active sources in an acoustic scene [1]. One of the most commonly used approaches is based on the Time-Difference-of-Arrival (TDoA) method, which is a two-step procedure. In the first step, one or several time delays between different pairs of microphones (i.e., the TDoAs) are estimated. The source position is determined in the second step using the array geometry and TDoAs. Well-known methods in this category are *Generalized Cross-Correlation* (GCC) and variants [2, 3]. In [4], the effects of reverberation on the TDoA estimates of GCC have been discussed in detail. It was found in the study that when the reverberation time reaches a certain threshold, the method becomes completely useless. This threshold was found to be $T_{60} = 600$ ms.

Other methods use frequency-averaged signal power of a *Steering Beamformer* (SB), where a steered response is generated by steering the beamformer over a predefined spatial region. A method combining both features of SB with the ones used for PHAse Transform (PHAT) weighting of the GCC is known as SRP-PHAT [5]. One inherent problem in all these methods is that they do not take into account any speech related property of the source, which makes them more vulnerable to short and abrupt acoustic events such as closing of a door and window or any spatially present non-speech source generating noise in the room. A new, joint pitch and position extraction algorithm, known as Joint-Position-Pitch (PoPi) algorithm [6], has been presented recently. The term *position* will be referred to as Direction-of-Arrival (DoA) for the rest of the paper.

The PoPi algorithm combines the DoA estimates with speaker-dependent features, which are instantaneous and therefore require no prior knowledge or model training. One of the most obvious speaker-dependent features is the fundamental frequency F_0, which is also referred to as *pitch*. The PoPi algorithm allows an acoustic source indexing in a multi-source environments, but tends to degrade for concurrent speaker cases. To represent multi-speaker scenarios, we further enhanced this method by including a pre-processing block inspired by the *auditory model* [7]. This led to the formulation of the Multi-band Position-Pitch (M-PoPi) algorithm. Preliminary results applying the M-PoPi algorithm have been reported in [8]. There, it was tested only on different vowels utterances by two speakers.

This paper takes the method a step further by presenting a generic framework combining M-PoPi algorithm with particle filters to handle different kinds of acoustic environments and speaker combinations. The method is tested on real speech signals of up to 3 concurrent speakers recorded in a room with reverberation time of $T_{60} = 650$ ms. Due to the challenging nature of the acoustic environment, the GCC methods for localization were not considered further in this paper. Whereas the SRP-PHAT algorithm has shown good and robust performance in most acoustic real-world conditions [5]. Therefore in this paper, we have compared our proposed method performance with the SRP-PHAT algorithm at two different levels. Firstly the M-PoPi and SRP-PHAT algorithms results are compared without employing particle filtering and then the comparison of both algorithms using particle filtering is carried out. Our proposed method performs well at both levels.

The rest of the paper is structured as follows: Section 2 gives a description of the M-PoPi algorithm with several possible variants and their applicability in different scenarios. In Section 3, an introduction of the particle filtering framework with new likelihood function based on the output of M-PoPi algorithm is proposed. Section 4 discusses the experimental framework followed by results and analysis in Section 5. Finally, Section 6 draws some conclusions and outlines future works.

[*] This research was carried out in the context of AAP-COMET, a joint project of Graz University of Technology, Philips Speech Processing, AKG Acoustics, and ATRONIC Austria.

2. THE M-POPI ALGORITHM

Fig. 1 shows a complete scheme outlining the M-PoPi algorithm. It is an extended version of the PoPi algorithm. A set of preprocessing modules are taken into consideration before computing the PoPi decomposition to overcome the inability of the basic algorithm in visualizing more than one active speaker in case of concurrent speakers.

The PoPi algorithm is used to extract the common periodicities that are present in multi-channel audio in addition to the cross-channel delay related to those periodicities from the cross-correlation function. This leads to the parameterized sampling of the cross-correlation function. The resulting position-pitch relations can be represented in a plane, the so-called PoPi plane, that reveals the peaks at locations that corresponds to joint position-pitch estimates of the active sources in an acoustic scene. A major drawback of the original PoPi formulation is that it tends to show the dominant speaker in case of multi-speaker scenarios. The dominant source in a speech mixture is the one with the highest energy resulting in the strongest peak in the cross-correlation function, whereas the weak source is not strongly present in the cross-correlation function. This results in poor PoPi decomposition leading to either one or both estimates of pitch and position to be incorrect. Fig. 2 presents the resulting PoPi planes for M-PoPi and standard PoPi algorithm in case of a multi-speaker scenario where two female speakers are active in a single frame.

2.1. The Gammatone Filterbank

The first module is inspired from the human cochlear model, which is implemented using a filterbank consisting of 64 overlapping bandpass gammatone filters, with center frequencies spaced uniformly on the equivalent rectangular bandwidth (ERB) scale between 50 Hz and 8000 Hz.

2.2. The Generalized Cross-Correlation (GCC)

In the next step, the cross-correlations are computed between a pair of microphone signals as such: first each signal is passed through a gammatone filterbank resulting in 64 output channels. Then the cross-correlations are computed between all the 64 outputs of both signal as shown in Fig. 1.

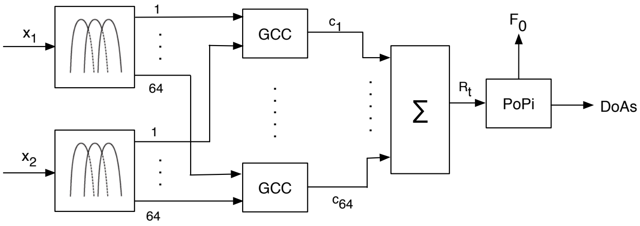


Figure 1: Block diagram of the M-PoPi algorithm for a single pair of microphones with each signal passing through a set of gammatone filterbank and then Generalized Cross-Correlation (GCC) calculated for every output channel of filterbank for one of the microphone signal with the corresponding filterbank outputs of the other microphone signal. The GCC functions are then summed to generate a mean cross-correlation function. The PoPi decomposition is applied on the mean cross-correlation function. For a multi-microphone pair system. All the PoPi planes are added together to create the final position and pitch estimates for the active sources in an acoustic scene.

$$R(\tau) = \int_{-\pi}^{\pi} W(\omega)\, X_1(\omega)\, X_2^*(\omega)\, e^{j\omega\tau}\, d\omega, \tag{1}$$

where $X_1(\omega)$ is the Fourier transform of $x_1(t)$ and $X_2^*(\omega)$ is the complex conjugate of the Fourier transform of $x_2(t)$, which is weighted by a weighting function, $W(\omega)$ and τ is the discrete time-lag.

Different weighting functions can be used depending on acoustic conditions. The PHAT weighting is particularly advantageous for high Signal-to-Noise Ratio (SNR) and reverberant scenarios. Whereas the Maximum Likelihood (ML) weighting can be used in cases where the noise statistics can be easily measured or is known apriori. When pitch is also computed along side the DoA estimates, the weighting function needs to be selected carefully. In case of PHAT, the cross-correlation function loses its periodicity, which holds information for the pitch estimation. This makes it unsuitable for the PoPi algorithm, but we can benefit from it's advantages by replacing the central part of cross-correlation function carrying the DoA information with the central part of the GCC-PHAT, where the correlation lag τ corresponds to $0° - 180°$. Hence the modified correlation function is given as:

$$R(\tau) = \begin{cases} R_{PHAT}(\tau), & \text{if } \tau \,\epsilon\, \langle \tau_{0°}, \tau_{180°} \rangle; \\ R_{CC}(\tau), & \text{otherwise.} \end{cases} \tag{2}$$

This way $R(\tau)$ keeps the advantages of GCC-PHAT, while maintaining the periodicity.

The summarization module normalizes the cross-correlation functions before the PoPi decomposition step. With the normalization, the relative information content of the multiple correlation functions is adequately represented and allows combinations of the information from these functions in a better way. With this step, the relative delays associated with position-pitch of all sources will be more enhanced in the case of multiple speakers reducing the impact of sources with higher SNR values. These multiple normalized cross-correlations are then summed up to form a mean cross-correlation function. We have used the mean cross-correlation function for the PoPi decomposition.

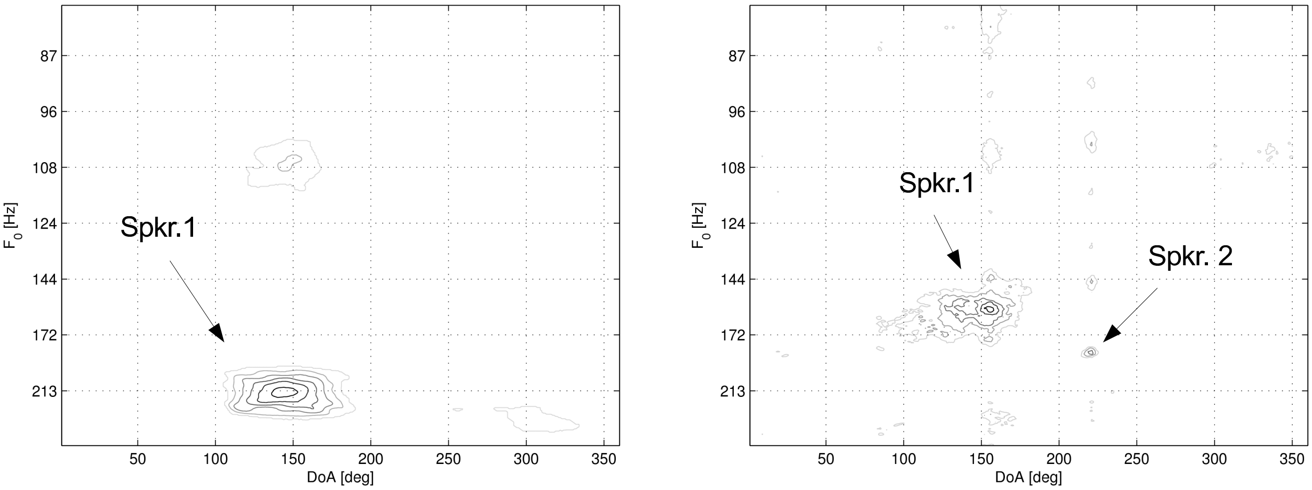
2.3. The Position-Pitch (PoPi) Algorithm

To evaluate the presence of a periodic signal with unknown fundamental frequency F_0, related to a source position at φ_0, the mean cross-correlation is then sampled accordingly:

$$\rho(\tau, F_0) = b \cdot \sum_{k=-K}^{K} R(\lfloor k \cdot L(F_0) \rfloor + \tau). \tag{3}$$

In this formulation, b denotes a normalization factor which is discussed later in the section, K defining the cross-correlation interval used for summation of samples. $L(F_0)$ being a first time-lag depending on the pitch parameter F_0 according to $L(F_0) = \frac{F_s}{F_0}$, F_s being the sampling frequency of the recorded signals. The time-lag value $k \cdot L(F_0)$ is being rounded using floor function to convert arbitrary real numbers that might result for the time-lag values to close integers.

The normalization factor b can be set equal to 1 or be used as the reciprocal of the number of correlation peaks considered. The



(a) PoPi

(b) M-PoPi

Figure 2: PoPi and M-PoPi decomposition of the same speech segment with 2 concurrent female speakers (Spkr. 1: $\varphi_0 = 155°$, $F_0 = 149$ Hz; Spkr. 2: $\varphi_0 = 221°$, $F_0 = 188$ Hz) using 24 channel circular microphone array. The M-PoPi algorithm correctly estimates both DoA and F_0 of each speaker, whereas the standard PoPi algorithm tend to show only one speaker with wrong DoA and F_0 estimate.

summation runs over a symmetric interval from $-K$ to K but it can also be ran from a specific correlation peak $-K_1$ to K_2. In this paper, we have used $b = \frac{1}{2K+1}$, where K is set to 3.

The sampling function generates time-lag value τ and fundamental frequency F_0, which corresponds to the active speaker. The source position φ_0 can be determined from τ by relation $\varphi_0 = cos^{-1}(\frac{\tau \cdot c}{d \cdot F_s})$, where c is the speed of sound in air and d is the distance between a pair of sensors.

In practice, the PoPi plane is evaluated only for predefined values of $L(F_0)$ and $\tau = O(\varphi_0)$, which are precalculated for the frequencies $F_0 = [80 \ldots 600]$ Hz and DoA candidates $\varphi_0 = [0° \ldots 180°]$ with a stepsize of $1°$.

For a microphone array with more than two sensors such as a uniform circular array (UCA), the position-pitch are estimated for pairs of oppositely placed microphones and then summed up. In order to cover a $360°$ view, the $0° - 180°$ response of each oppositely placed microphone pair was first mirrored around its axis before the summation.

3. PARTICLE FILTER FRAMEWORK FOR SOURCE LOCALIZATION

The sequential Monte Carlo methods, commonly known as particle filters, are widely used in practical applications of tracking single and multiple speakers due to their ability in dealing with multimodality, non-linear functions and non-Gaussian noise. The particle filters are state-space based approaches based on the key idea that the peaks due to true sources follow a dynamical model from frame to frame whereas there is no temporal consistency present in the outliers.

The tracking problem can be formulated in the following manner, let $\mathbf{y}_{1:t} = [\mathbf{y}_1. \cdots, \mathbf{y}_t]$ denote the concatenation of all measurements up to time t and the task at hand is to track source with source state defined as $\boldsymbol{\alpha}_t = [\hat{\varphi}_1, \hat{\varphi}_2, \cdots, \hat{\varphi}_K, T_K]$, where $\hat{\varphi}_k$ is the DoA for source k and T_K is the total number of sources active at the current time-step t. The aim is then to recursively estimate the posterior filtering distribution $p(\boldsymbol{\alpha}_t|\mathbf{y}_{1:t})$ using Bayes' Theorem as follows:

$$p(\boldsymbol{\alpha}_t|\mathbf{y}_{1:t-1}) = \int p(\boldsymbol{\alpha}_t|\boldsymbol{\alpha}_{t-1})\, p(\boldsymbol{\alpha}_{t-1}|\mathbf{y}_{1:t-1})\, d\boldsymbol{\alpha}_{t-1}$$
$$p(\boldsymbol{\alpha}_t|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\boldsymbol{\alpha}_t)\, p(\boldsymbol{\alpha}_t|\mathbf{y}_{1:t-1}). \tag{4}$$

The first step is the *prediction step*, which will use the combined dynamical model $p(\boldsymbol{\alpha}_t|\boldsymbol{\alpha}_{t-1})$, to propagate the previous posterior, $p(\boldsymbol{\alpha}_{t-1}|\mathbf{y}_{1:t-1})$, to give the estimate of the predictive distribution $p(\boldsymbol{\alpha}_t|\mathbf{y}_{1:t-1})$. The second step is the *update step*, where the likelihood, $p(\mathbf{y}_t|\boldsymbol{\alpha}_t)$ is combined with the predictive distribution at time-step t.

Particle filters essentially implements the recursions in (4) by using a large set of discrete samples, or particles, with associated discrete probability masses commonly known as weights. In order to combine the M-PoPi algorithm with particle filters, there are three building blocks to select such as: the dynamic model, the localization function, and the likelihood function.

Keeping in view the current problem of speaker localization ranging from one up to multiple speakers present at static position. The main steps of particle filtering method are outlined as follows:

1. **Initialization of Particle Filters**: The particle filters are randomly distributed in the state-space, ${\boldsymbol{\alpha}_0}^i$ with associated uniform weights ${w_0}^i = 1/N, i = 1 : N$. In our experiments, we have chosen $N = 100$.

2. **Dynamical Model**: Predict the new set of particles according to Langevin dynamics model with similar settings as used in [9].

3. **Localization Function**: To transform the raw data received at the sensors into localization measurements, we use the

M-PoPi algorithm output maximized along the DoA dimension.

4. **Likelihood Function**: The M-PoPi algorithm is used as a *pseudo-likelihood* function $F(\mathbf{y}_t, \boldsymbol{\alpha})$. The details for this function are summarized in section 3.1. The new weights corresponding to particles are assigned to as:

$$w_t{}^i = p(\mathbf{y}_t|\boldsymbol{\alpha}_t{}^i), \tag{5}$$

and normalized to obtain $\sum_{i=1}^{N} w_t{}^i = 1$.

5. **Resampling**: Resample the particles by multiplying the particles with higher weights and deleting the ones with smaller weights to avoid the degeneracy problem using a suitable resampling method. Systematic resampling is used in our framework and weights are reset to uniform values.
6. **Location Estimation**:The final estimate for the location of sources can be calculated by clustering the particles' set or a histogram measure using a carefully selected threshold.

3.1. M-PoPi based Likelihood Function

The likelihood function should be chosen to reflect that the peaks in the localization function belong to likely source positions. Additionally it should also refelect the fact that there might be no peak belonging to any source locations such as when no source is active or the presence of spurious peaks due to background noise and sensor calibration errors.

The *pseudolikelihood* functions derived from M-PoPi algorithm output based on the formulation of [9] is given as:

$$F(\mathbf{y}_t, \boldsymbol{\alpha}) = \max\{\mathbf{y}_t(\hat{\varphi}_{\boldsymbol{\alpha}}), \xi_0\}^r, \tag{6}$$

where $\hat{\varphi}_{\boldsymbol{\alpha}}$ is the localization parameter corresponding to the state, $\xi_0 \geq 0$, and $r \in \mathbb{R}^+$. The use of r as explained in [9] is to help shape the localization function to make it more amenable to recursive estimation. The presence of ξ_0 ensures that the function is non-negative and includes the case where no peak in the localization function belong to the true source. The likelihood function used to assign new weights to the particle filters is then calculated as, $p(\mathbf{y}_t|\boldsymbol{\alpha}_t{}^{(i)}) = F(\mathbf{y}_t, \boldsymbol{\alpha}_t^{(i)})$. In our experiments, we used the values of $\xi_0 = 0$ and $r = 2$.

3.2. Particle Filter with Integrated Voice Activity Detection

During the silence periods occurring in the middle of speech signals, the tracking algorithm keeps on updating the source locations as if the source was still active. To mitigate this problem, voice activity detection should be included in the tracking framework. We have followed the idea introduced in [10] to integrate voice activity detector in the likelihood function such as:

$$p(\mathbf{y}|\boldsymbol{\alpha}) = q_0 \cdot \mathcal{U}_{\mathcal{D}}(\hat{\boldsymbol{\varphi}}_{\boldsymbol{\alpha}}) + \gamma \cdot (1 - q_0) \cdot Po(\hat{\boldsymbol{\varphi}}_{\boldsymbol{\alpha}}), \tag{7}$$

where the subscript t has been omitted for sake of simplicity. The value q_0 represents the hypothesis that the measurement originates from clutter, and $1 - q_0$ indicates that the measurement originates from true source. And $\hat{\boldsymbol{\varphi}}_{\boldsymbol{\alpha}}$ corresponds to the state vector $\boldsymbol{\alpha}$ and with $\mathcal{U}_{\mathcal{D}}$ the uniform PDF over the considered state-space $\mathcal{D}$. The second term in the equation is the *pseudo-likelihood* function $Po(\cdot)$ derived from the M-PoPi algorithm as explained in previous section with the normalization constant γ ensuring that this function is suitable for use as a density function.

During the silence periods, this integration allows the tracking algorithm to put more emphasis on the considered dynamics model in spreading the particles, while at the same time reducing the importance of M-PoPi observations due to the fact that no useful information is present when speaker is inactive. This allow the particle filter to keep track of silent speaker and resume tracking successfully when the speaker becomes active again.

3.3. Additional Module for Removal and Addition of Particles

The performance of the tracking algorithm was further improved by creating a heuristic approach, where a certain percentage of particles is deleted from the set at every iteration. After several trials, a value of 20% was selected as it exhibits the best performance. The deleted particles were replaced by a newly propagated set of particles that were randomly placed in the neighborhood of the peaks obtained by a peak-picking algorithm with maximum of 5 peaks selected at every iteration. The corresponding weights were assigned using the *pseudolikelihood* function.

This step was carried out for both M-PoPi and SRP-PHAT based particle filtering algorithms. This additional step reduced the effect of a higher weighting imposed by the tracking algorithm on the dominant speaker location relative to the weak speaker location. This is especially helpful for multi-speaker scenarios, where one speaker due to higher SNR will be more strongly present in the speech mixture than the others with low SNR values.

4. EXPERIMENTAL FRAMEWORK

The performance of the algorithm was evaluated on data recorded using 7 Yamaha MSP5A loudspeakers and a 24 channel UCA in the SPSC meeting room. This meeting room has the dimensions $6.02 \times 5.32 \times 3$ m and a reverberation time $RT_{60} = 650$ ms. One of the walls of the room has a large window partly covered by blinds that were set open during the recordings. The floor is covered with standard carpet. No particular effort was made to reduce the reverberations in the room.

The array has been designed with 24 Behringer ECM8000 omni-directional microphones positioned equidistantly with an inner diameter of 55 cm on a circular ring connected to an M-Audio Firewire Audiophile Mobile Recording Interface under control of a laptop computer.

For the recording a subset of Keele [11] and MOCHA-TIMIT [12] databases was used. A set of 7 speech files containing 3 male and 4 female utterances were mixed into longer segments modeling different speaker interaction behaviors in a spatialized multi-speaker scenarios. The array was placed in the center of the room; the loudspeakers were positioned at a height of 1.39 m maintaining a constant distance of approx. 2 m from the array. The azimuths of all speakers with their respective gender tags are outlined in Table. 1. The playback and recording process was controlled by software on a single laptop and the captured audio was saved directly to the hard disk of the laptop with 16 bit resolution and sample rate of 48 kHz.

To evaluate the localization performance of the algorithms, a frame level metric, denoted as *Acc*, is used. This measure has also been used in [13] for the localization estimate. The measure is based on the normalized number of frames, (where the estimated localization angle $\hat{\varphi}_n$, is close enough to the true angle φ_0, to be considered correct $\hat{\varphi}_n$ is scored as correct if it is close to the true

DoA	S1(F)	S2 (F)	S3 (M)	S4 (M)	S5 (F)	S6 (F)	S7 (M)
φ_0	221°	155°	122°	248°	298°	342°	15°

Table 1: Azimuths of speakers (S1-S7) with 3 male (M) and 4 female (F) speakers.

	S1	S2	S3	S4	S5	S6	S7	Mean	S1+S2	S1+S2+S3
SRP-PHAT	65.2	88.8	67.2	39.6	84.8	94.4	90	75.7	54.5	43.7
MPoPi	85.8	94.7	86.3	71.4	97.5	**97.8**	96.2	90	58.8	52.5
SRP-PHAT-PF	**96**	**99.7**	74.5	92.6	98	97.7	99	93.9	57.5	49.4
MPoPi-PF	92.7	**99.7**	**87**	**97**	**98.6**	97.7	**100**	**96**	**71.5**	**60.5**

Table 2: Localization accuracy in percent for all three speaker configuration, where bold values represent the best performance achieved out of all 4 algorithms for every case.

angle of any of the speaker):

$$Acc = \frac{1}{N}\sum_{n=1}^{N}\delta^*(\varphi_0, \hat{\varphi}_n) \times 100\% \tag{8}$$

where N is the number of frames. δ^* is defined as

$$\delta^*(a,b) = \begin{cases} 1 & \text{if } |a-b| \le \Delta \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

where Δ is a grace boundary around the true angle within which the estimated angle is considered to be correct. The value of grace boundary was fixed at 5°. This correspond to the minimal inter-speaker distance of 35 cm in a 2 m distance from the array.

5. RESULTS AND ANALYSIS

A comparative analysis between the M-PoPi and SRP-PHAT algorithms for single up to three active speaker scenarios is carried out at two levels. Firstly both algorithms have been tested without using the tracking framework, and then the particle filtering framework similar to the M-PoPi algorithm is used in conjunction with SRP-PHAT algorithm, where the output of SRP-PHAT is used as the *pseudolikelihood* function. And both algorithms are again compared by utilizing the particle filtering framework. The methods are evaluated with a frame length of 104.2 ms with a frame shift of 52.1 ms for all recordings.

For SRP-PHAT, the functions were computed only for azimuths over a range of $0^\circ - 360^\circ$ with a resolution of 1°. No comparative study was undertaken with the original PoPi algorithm, as it tends to show either wrong estimates of position and of pitch of the sources or only the dominant speaker, as illustrated in Fig. 2 .

Tab. 2 shows the averaged frame correctness scores of the M-PoPi and SRP-PHAT algorithms for all 7 single speakers and up to 3 concurrent speakers. The cases of concurrent speakers were made with different combinations of speakers. The 2 Speaker case was made with S1 and S2 from the azimuth table consisting of two female speakers and in 3 speaker case, S3 a male speaker was added. Due to relatively strong presence of some sources to the others, an averaged frame correctness score was used instead of individual score of each speaker. It also makes the comparison of two methods more straightforward.

As shown in Tab. 2, the proposed method performs better in comparison to the SRP-PHAT method for all speaker cases. In case of single speakers, both algorithms performance improve using particle filters. The results for the original algorithm for one source scenario are varying among different speakers. It was observed that the speakers with low SNR have least frame correctness score, e.g, S1 had SNR of around 15 dB with only 65% correctness score for SRP-PHAT. In this case the M-PoPi algorithm still delivers an average score of 85%. On the other hand speaker S5 to S7 had SNR of more than 20 dB, where SRP-PHAT algorithm detects 90% frames correctly with M-PoPi delivering 97% frames correctly without the use of particle filters.

The use of particle filters is more apparent for the weak sources, as with particle filters both algorithm have more than 90% frame correctness score for all single speakers. This highlights the importance of using a post-processing stage, which works under the principle that the true sources follow a dynamical model, whereas false sources that appear due to background noise and multipath propagation have no temporal continuity. The particle filtering algorithm has the most significant effect on S4, which had really poor results for detection only algorithms but produced around 97% score for M-PoPi based particle filters and 92.56% for SRP-PHAT based particle filters. This highlights that the framework defined with M-PoPi algorithm performs better on average than the one using SRP-PHAT for all seven speaker cases.

The concurrent speaker case, where in a single analysis frame multiple speakers are present poses a challenging problem. Both localization algorithms give rather poor results for each case with M-PoPi giving an absolute improvement of 4.3% and 8.77% over SRP-PHAT algorithm for 2 and 3 concurrent speaker cases respectively. The use of particle filtering method with M-PoPi algorithm improves localization accuracy by 12.7% and 8% in comparison to M-PoPi method. The SRP-PHAT based particle filtering method only improves the performance by 2.98% and 5.63% for the same two cases. It can be clearly seen from these results that the performance of particle filters depends upon the localization function. The SRP-PHAT method computes the averaged power response, which directly depends upon the relative loudness of all speakers. As the weak sources are dominated by strong sources with higher SNR values. Though the localization accuracy of M-PoPi also degrades for weak sources but the affects of relative SNR values are much less pronounced than SRP-PHAT algorithm.

The advantage of using F_0 as an additional feature for speaker discrimination in M-PoPi algorithm is clearly visible, and F_0 had a much significant effect on localization performance as the number of speakers increases.

6. CONCLUSIONS

To improve the localization of single to multiple active sources in an acoustic environment, the M-PoPi algorithm has been combined with particle filters and a new likelihood function has been proposed. The proposed method has been tested on multi-channel speaker recordings using a circular microphone array in a highly reverberant meeting room with background noise. The results have been compared to the state-of-the-art SRP-PHAT algorithm. It shows better performance for all speaker combinations ranging from single up to three concurrent speakers. The M-PoPi algorithm has proved to be a better localization and likelihood function than SRP-PHAT algorithm as an average relative gain in localization accuracy of more than 10% is achieved over the SRP-PHAT algorithm. This improvement results partly from assigning pitch value to the position of the sources and partly from introducing a pre-processing stage with application of multiple sensor pairs. Future work will focus on moving concurrent source scenarios.

7. REFERENCES

[1] M. S. Brandstein and D. B. Ward, *Microphone Arrays: Signal Processing Techniques and Applications*, Springer, 2001.

[2] C. F. Knapp and G. C. Carter, “The generalized correlation method for estimation of time delay,” *IEEE Trans. on Acoustic, Speech and Signal Processing*, 1976.

[3] M. Omologo and P. Svaizer, “Acoustic source location in noisy and reverberant environment using csp analysis,” in *Proc. of ICASSP*, Atlanta, 1996.

[4] B. Champagne, S. Bédard, and A. Stéphenne, “Performance of time-delay estimation in the presence of room reverberation,” *IEEE Trans. on Speech and Audio Processing*, vol. 4, no. 2, 1996.

[5] J. H. Dibiase, *A high accuracy, low latency technique for talker localization in reverberant environments using microphone arrays*, Ph.D. thesis, Brown University, 2000.

[6] M. Képesi, F. Pernkopf, and M. Wohlmayr, “Joint position pitch tracking for 2-channel audio,” in *International Workshop on Content based Multimedia Indexing*, Bourdeaux, France, June 2007.

[7] M. Slaney, “Auditory toolbox: A matlab toolbox for auditory modeling work,” Tech. Rep. 45, Apple Computer, Inc. Advanced Technology Group, 1994.

[8] M. Képesi, L. Ottowitz, and T. Habib, “Joint positon-pitch estimation for multiple speaker scenarios,” in *IEEE Workshop HSCMA*, Trento, Italy, 2008.

[9] D. B. Ward, E. A. Lehmann, and R. C. Williamson, “Particle filtering algorithms for tracking an acoustic source in a reverberant environment,” *IEEE Trans. on Speech and Audio Processing*, vol. 11, 2003.

[10] E. A. Lehmann and A. M. Johansson, “Particle filter with integrated voice activity detection for acoustic source tracking,” in *EURASIP Journal on Applied Signal Processing*, 2007.

[11] G. F. Meyer, F. Plante, and W. A. Ainsworth, “A pitch extraction reference database,” in *Proc. of Eurospeech*, Madrid, Spain, Sept. 1995.

[12] A. Wrench, “The mocha-timit articulatory database,” http://www.cstr.ed.ac.uk/artic/mocha.html, Queen Margaret University College 1999.

[13] H. Christensen, N. Ma, S. N. Wrigley, and J. Barker, “A speech fragment approach to localising multiple speakers in reverberant environments,” in *Proc. of ICASSP*, Taipei, Taiwan, Apr. 2009.