Audio Engineering Society

Convention Paper 8157

Presented at the 129th Convention
2010 November 4–7 San Francisco, CA, USA

Calculating time delays of multiple active sources in live sound

Alice Clifford[1], Josh Reiss[1]

[1] *Centre for Digital Music, Queen Mary, University of London, London, E1 4NS, UK*

Correspondence should be addressed to Alice Clifford (alice.clifford@eecs.qmul.ac.uk)

ABSTRACT

Delays caused by differences in distance between sources and microphones cause many problems in live audio, most notably comb filtering. This paper presents a new method that is able to calculate the relative time delays of multiple active sources to multiple microphones where previous methods are unable to. The calculated time delays can be used to compensate for delays that cause comb filtering and can also be used in source separation methods which utilise delays. The proposed method is shown to be able to calculate delays in configurations where other methods fail and is also able to give an estimate of sources physical positions. The results show that multiple delays can be accurately calculated when multiple sources are active and that noise can effect the accuracy of the method.

1. INTRODUCTION

Common practice in both live sound and studio recording is to record a single sound source with multiple microphones. Sound radiates from an instrument in all directions but the sound picked up by a microphone differs depending on the microphone's position around the instrument. For example the sound from the strings of a guitar sound very different to the sound resonating from the body. For this reason multiple microphones can be used to reproduce different qualities of an instrument and can be mixed together to create the desired sound.

It is difficult, and sometimes not desired, to place these microphones equidistant from the main source. If this is the case the sound from the instrument will arrive at each microphone with different delays. When the microphone signals are mixed microphone artefacts can occur, for example comb filtering which changes the frequency content of the signal and is generally undesired.

Differences in source to microphone delays can also occur when multiple microphones are used to repro-

duce multiple sources, for example in an ensemble performance where each instrument has a dedicated spot microphone. Microphone bleed can occur between the microphones and can also cause comb filtering if mixed. Similar problems can occur when a stereo microphone pair is used to reproduce an ensemble of instruments and the instruments have their own dedicated microphones. The sound from an instrument will arrive at the spot microphone and the stereo pair with different delays. With a large ensemble, many delays can occur.

If the difference in the delay of a source arriving at multiple microphones can be calculated, manual delay can be applied to the microphone signal with the shortest delay. Both sources would then have equal delay. This can be achieved in live sound by either calculating the delays that are occurring by measuring microphone and source positions or by applying delay "by ear" until the comb filtering has been reduced. It is now possible to use signal processing to automatically estimate the delays using a time delay estimation (TDE) method with no knowledge of the microphone or source positions [1], the most common being the Generalized Cross Correlation (GCC) [2], which calculates the difference in delay of a single source to two microphones. Weightings can also be applied to improve the performance of the GCC in noisy and reverberant conditions. An example of this is the Phase Transform (PHAT). A system for estimating the relative delays of a single source arriving at multiple microphones and automatically applying the correct delay compensation is presented by Perez Gonzalez and Reiss in [3].

The GCC-PHAT method is also used in source separation, for example in [4]. Source separation attempts to isolate sources from a mixture by estimating the mixing parameters, usually delay and gain, of each source and using these to create unmixing filters. The sources can then be kept separate or processed separately and mixed back together.

The DUET method of source separation of mixed signals [5] calculates the delay parameters using a different method. This method estimates the phase difference for each frequency bin and histograms the result. An estimate of the amplitude of each bin is also included to produce peaks in the histogram. The position of these peaks determines the attenuation and delay of each source. The number of peaks is equal to the number of sources. Unlike most source separation methods, this does not use GCC for the delay estimation.

The DUET method is able to calculate the relative delays of multiple sources to multiple microphones, but it relies on the input sources having W-disjoint orthogonality, meaning they do not overlap in time and frequency at a given time. It is also very sensitive to noise and reverberation, which effects the quality of the source separation. The DUET method also requires that the microphones be close together and it is only useful for 2 microphones. This is because the distance between the microphones is determined by the highest frequency in the audio sample. If the highest frequency is assumed to be 16kHz there can be a maximum distance of 2.15cm [6], which is a significant constraint, especially if estimating delays of spot microphones as instruments will be placed much further apart.

This paper proposes a new method for calculating the relative delays of multiple sources to multiple microphones using the with GCC-PHAT which allows for much wider microphone spacing and does not require W-disjoint orthogonality of sources.

2. SINGLE ACTIVE SOURCE

A simple configuration of a single source being reproduced by multiple microphones is described by

$$\begin{aligned} x_1[n] &= s[n - \tau_1] \\ x_2[n] &= s[n - \tau_2] \end{aligned} \tag{1}$$

where the relative delay is defined as

$$\tau_s = \tau_2 - \tau_1 \tag{2}$$

assuming

$$\tau_2 > \tau_1 \tag{3}$$

2.1. Time Delay Estimation

The GCC time delay estimation technique [2] is a method of performing cross correlation in the frequency domain and is able to calculate τ_s. It is defined by

$$\Psi_G[n] = \mathcal{F}^{-1}\left\{X_1^*[k] \cdot X_2[k]\right\} \tag{4}$$

for frequencies $k = 0, ..., N-1$ where N is the analysis window size. $\mathcal{F}^{-1}$ denotes the inverse Fast Fourier Transform, X_1 and X_2 are the microphone signals x_1 and x_2 in the frequency domain and $(\cdot)^*$ denotes the complex conjugate. The delay, τ_s is calculated by the position of a single maximum peak in the GCC function. τ_s is then calculated by

$$\tau_s = \arg\max_n \Psi_G[n] \tag{5}$$

This method is popular due to its simplicity and its ability to be weighted to improve the estimation accuracy in noisy and reverberant environments. For example applying the Phase Transform (PHAT) weighting when performing the GCC sets all frequency magnitudes equal to 1, preserving the phase information. This changes the GCC to

$$\Psi_P[n] = \mathcal{F}^{-1}\left\{\frac{X_1^*[k] \cdot X_2[k]}{|X_1^*[k] \cdot X_2[k]|}\right\} \tag{6}$$

being referred to as the GCC-PHAT. The results can also be improved by using an N-point Hann window to window the microphone signals before performing the FFTs. The output of a GCC-PHAT calculation can be seen in Fig. 1 where the horizontal position of the peak determines the estimated delay.

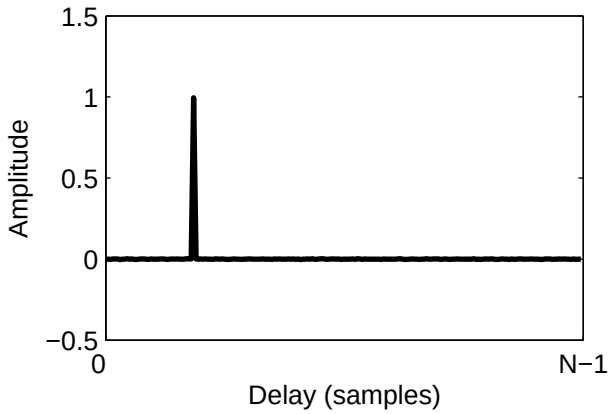

Fig. 1: Output of the GCC-PHAT

The single peak GCC-PHAT can estimate relative delays up to $N/2$ where N is the window size. With a sampling rate of 44.1kHz and a window size of 2048 samples, which is used in this paper, this is equal to a delay of 0.0232s. Taking the speed of sound as 343m/s at 20°C, this is equal to a maximum distance between microphones of 7.95m when a source is at a ±90° angle. In reality, the distance can be larger than this if a source is positioned further in front of the microphones. The distance can also be increased by increasing the window size.

A similar method to the GCC shown in [7] calculates the transfer function between the two signals and finds the subsequent maximum peak in the time domain. This is defined by

$$\Psi_I[n] = \mathcal{F}^{-1}\left\{\frac{X_2[k]}{X_1[k]}\right\} \tag{7}$$

It was found by the authors that if the PHAT is applied to Eq. (7) it becomes equal to the GCC-PHAT as the magnitude no longer effects the calculation, being made equal to 1. The equality can be shown by analysing how the phase of the signals change. Taking the calculations before the inverse FFT in Eq. (4) and Eq. (7) and analysing the phase gives

$$\arg(X_1^*[k]X_2[k]) = \arg\left(\frac{X_2[k]}{X_1[k]}\right) \tag{8}$$

The rest of this paper will be concerned with the GCC-PHAT method as it is straightforward to implement and is easily manipulated [1].

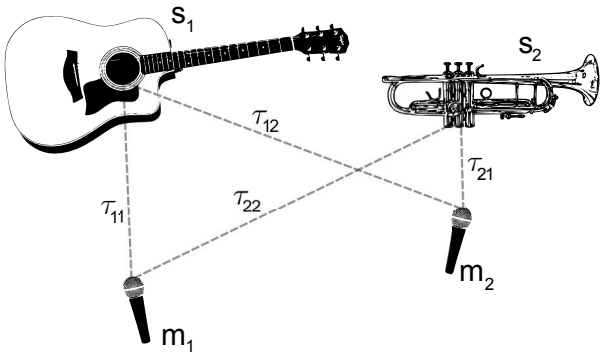

Fig. 2: Real life example of Equation (9)

3. MULTIPLE ACTIVE SOURCES

In some live sound configurations there may be multiple sources being reproduced by multiple microphones, for example in the ensemble example mentioned previously. Taking the two source, two microphone case, if all sources are active, i.e. producing sound, the configuration is described as

$$\begin{aligned} x_1[n] &= s_1[n-\tau_{11}] + s_2[n-\tau_{21}] \\ x_2[n] &= s_1[n-\tau_{12}] + s_2[n-\tau_{22}] \end{aligned} \tag{9}$$

A real life configuration of this is shown in Fig. 2. Eq. (9) assumes that

$$\begin{aligned} \tau_{11} &< \tau_{21} \\ \tau_{22} &< \tau_{12} \end{aligned} \tag{10}$$

where s_1 is placed closest to x_1 and s_2 is placed closest to x_2. The difference in delay of s_1 and s_2 to microphones x_1 and x_2 is described as.

$$\begin{aligned} \tau_1 &= \tau_{21} - \tau_{11} \\ \tau_2 &= \tau_{12} - \tau_{22} \end{aligned} \tag{11}$$

τ_1 and τ_2 are the differences in source to microphone delays that are desired and will be estimated using the proposed method.

3.1. GCC-PHAT

Eq. (9) can be viewed as two different instances of Eq. (1), a single source being reproduced by two microphones. As the single peak GCC-PHAT is used to find the relative delay of a single source to two microphones τ_1 and τ_2 from Eq. (11) can then be estimated with two single peak GCC-PHAT calculations, switching the microphone signals in Eq. (6) to become

$$\begin{aligned} \Psi_{P12}[n] &= \mathcal{F}^{-1}\left\{\frac{X_1^*[k]\cdot X_2[k]}{|X_1^*[k]\cdot X_2[k]|}\right\} \\ \Psi_{P21}[n] &= \mathcal{F}^{-1}\left\{\frac{X_1[k]\cdot X_2^*[k]}{|X_1[k]\cdot X_2^*[k]|}\right\} \end{aligned} \tag{12}$$

then

$$\tau_1 = \arg\max_n \Psi_{P12}[n] \tag{13}$$

$$\tau_2 = \arg\max_n \Psi_{P21}[n] \tag{14}$$

In this case, only one delay is estimated from each GCC-PHAT calculation. As a consequence of this the traditional single peak GCC-PHAT method is unable to estimate the relative delays of all sources if $N_s >= N_m$, where N_s is the number of sources and N_m the number of microphones, without manipulation of the output as the number of calculations that will be performed will be less than the number of delays to be calculated. The number of delays to be estimated in a N_s source, N_m microphone configuration is described by

$$N_d = N_s(N_m - 1). \tag{15}$$

The maximum number of calculations that could be performed by the single peak GCC-PHAT, estimating one delay from each calculation, is defined by

$$N_{Cs} = N_m(N_m - 1) \tag{16}$$

therefore if $N_s > N_m$ then $N_{Cs} < N_d$. The ability of the single peak GCC-PHAT to calculate multiple delays where $N_s < N_m$ is dependent on the position of the sources to the microphones and will not be discussed in this paper.

4. PROPOSED METHOD

This paper proposes a method whereby multiple delays can be estimated by using the same GCC-PHAT calculation, Eq. (6), but making use of redundant information usually ignored as the GCC-PHAT is designed for estimating the delay of a single source to multiple microphones. The proposed method is able to calculate relative delays for cases where $N_s >= N_m$, whereas the single peak method does not. The proposed multiple peak method also does not require W-disjoint orthogonality; both sources can be active, which is required in the DUET method, therefore they can be highly correlated and still the delays calculated.

The GCC-PHAT can only be calculated between two microphones. When there are N_m microphones and N_s sources, N_s peaks will appear in each calculation assuming all delays are unique. Each peak is equal to the relative delay of each source to the microphone signals under observation. The single peak GCC-PHAT discussed previously uses the delay estimated from the maximum peak only.

Fig. 3 shows the output of a GCC-PHAT calculation in the two source, two microphone case with the delays labelled. The estimation of the delays using multiple peaks from this calculation is described as

$$\tau_1 = arg\max_n \left[\Psi_P[0], \ldots, \Psi_P[\frac{N}{2}]\right] \quad (17)$$

$$\tau_2 = \frac{N}{2} - \arg\max_n \left[\Psi_P[\frac{N}{2}+1], ..., \Psi_P[N-1]\right]. \quad (18)$$

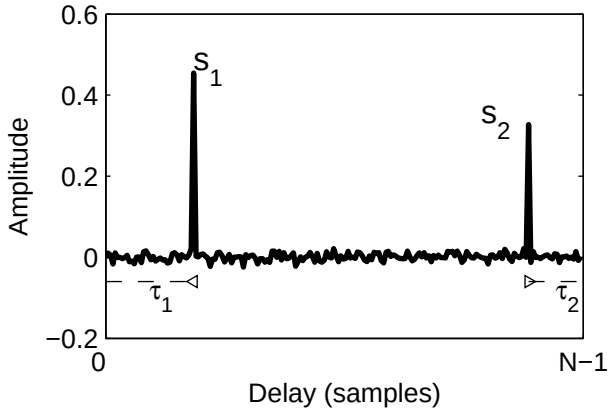

Fig. 3: Output of the GCC-PHAT where 2 sources are present with the delays labelled.

In the 2 source, 2 microphone case, using the multiple peak GCC-PHAT reduces the number of calculations per window from 2 to 1. As N_s and N_m increase, the number of maximum calculations that are performed with the multiple peak GCC-PHAT is

$$N_{Cm} = \frac{N_m(N_m-1)}{2} \quad (19)$$

which is still less than the maximum number of calculations for the single peak GCC-PHAT method, shown in Eq. (16), by a factor of 2.

The advantage of less calculations is that as the number of sources and/or microphones increases, the number of calculations inevitably increases with either method. The aim of this technique is to run in realtime as it is aimed at live sound. If the number of calculations are too many, a lot of processing power will be required to achieve realtime implementation, therefore less calculations is desired. If the delay estimation of longer delays is required, a longer window size is required which increases the processing required. If less calculations are required, this will decrease the processing time.

With the assumption shown in Eq. (10), the GCC-PHAT is used to calculate delays of up to $N/2$ where N is the window size. This means in the single peak method, Eq. (12), $\Psi_{P12}[n]$ and $\Psi_{P21}[n]$ are only used where $n = 0, \ldots, N/2$. It also happens that $\Psi_{P12}[n]$ where $n = N/2+1, \ldots, N-1$, which is unused, is equal to $\Psi_{P21}[n]$ where $n = 0, \ldots, N/2$ but with the elements reversed.

As the multiple peak method in this case would use $\Psi_{P12}[n]$ where $n = 0, \ldots, N-1$, both methods will estimate the delay from the same data, making the accuracy of each equal.

If the configuration is extended to the N_m microphone N_s source case the technique is the same as in Eqs. (17) and (18) but for N_s peaks. If a peak occurs in the first half of the function, the delay is calculated by Eq. (17), if it occurs in the second half, it is calculated by Eq. (18). This is repeated up to the number of sources.

One method for calculating the position of the peaks would be to read the positions one source at a time, removing the peak after the delay has been extracted continuing this up to the number of sources.

The multiple source GCC-PHAT provides other information about the sources. A peak that occurs at the 0 or N-1 position is cause by a source that is equidistant from both microphones. A peak that occurs in the first half of the output function is caused

by a source positioned to the left of the centre line between the microphones and a peak that occurs in the second half of the output will be caused by a source to the right.

The amplitude of the peak also determines the relative distance of each source to the microphones. The peak with the highest amplitude will be caused by a source placed closest to the microphones, the smallest caused by a source placed furthest away. This is shown in Fig. 4 where it can be seen that s_1 is closest to m_1, positioned to the left of the centre (dashed) line. This is shown in the GCC-PHAT function as s_1 appears in the first half of the function with a large amplitude.

After the multiple delays have been calculated, it is desirable to know which delays correspond to which sources. For this, a simple estimation of the relative placement of sources and/or distance from microphones is required to assign each estimated delay to the correct source.

4.1. Effect of noise

The accuracy of the delay estimation can be affected by additional uncorrelated noise such as that from other instruments, audience noise and air conditioners as this increases the noise floor leading to a decrease in the Signal-to-Noise ratio (SNR). The noise floor will eventually reach a similar level to that of the desired peak, thus making the peak difficult to find in the GCC-PHAT output. The noise floor also affects the effective distance the source can be from the microphone and the delay to still be estimated accurately. As a source moves from the microphones the amplitude decreases due to distance thus the amplitude of the peak caused by that source will have a smaller amplitude which may be less than the noise floor.

5. COMPARISON

Experiments were run to ascertain the robustness of the proposed method using multiple peaks to the traditional single-peak GCC-PHAT against noise and to confirm that the performance of the proposed method was equal to that of the single peak method. This was performed using a simulation whereby the position of microphones and sources were defined and delays and gain changes calculated. These variables were then used to construct simulated microphone signals. The input signals were a direct-input

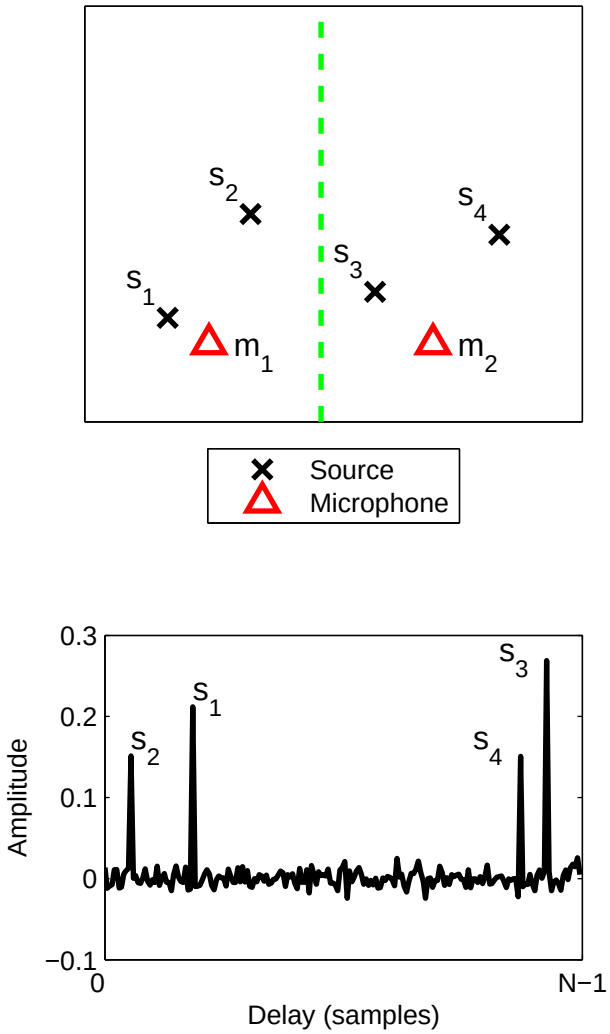

Fig. 4: Sample layout of sources and microphones (top) and the resulting GCC-PHAT function (bottom) showing how the amplitude and position of the peaks is related to the position of the sources.

recorded guitar and piano sample taken direct from the keyboard output therefore there was little noise or reverberation on the original samples. The layout of sources and microphones can be seen in Fig. 5.

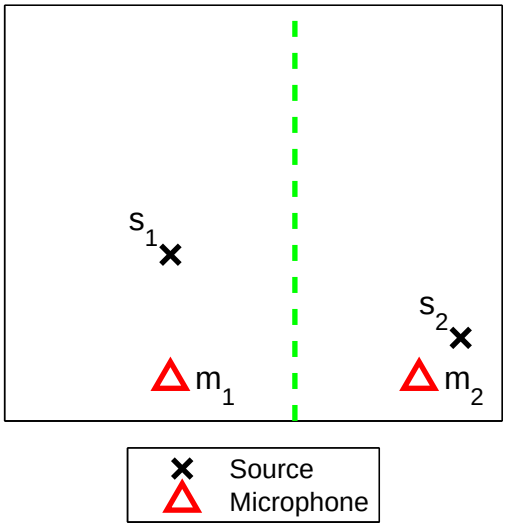

Fig. 5: Layout of sources and microphones for the test of robustness to uncorrelated noise.

Uncorrelated pink noise was then added to increase the SNR in dB of the source signal furthest from the microphone. The noise was added to alter the SNR in 0.5dB increments from 60dB to 0dB. Each SNR step was averaged 100 times.

5.1. Results

A 10 second sample was used of each input which was windowed with a 2048 sample Hann window. For each SNR step, the percentage of windows where the correct delay was estimated was calculated, which can be seen in Figure 6 which shows the results for the single and multiple peak GCC-PHAT. The results were equal.

As expected, the results are equal because the amplitude of the peak for each source will always be the same, the only difference being whether the peak is in the first half of the output or the last half. Although both results are equal, the multiple peak GCC-PHAT required half the calculations of the single peak GCC-PHAT.

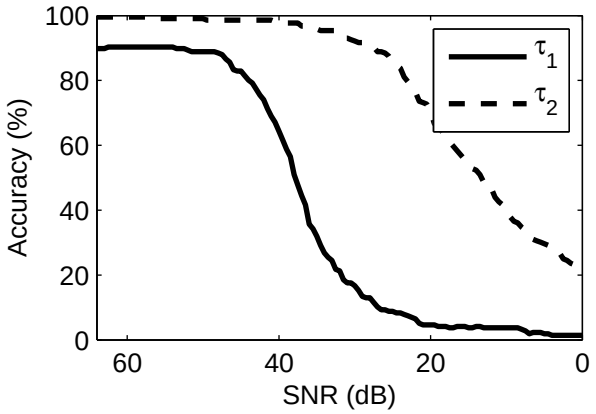

Fig. 6: Results of noise test for the single and multiple peak GCC-PHAT methods which were equal.

The results show that both the traditional single peak GCC-PHAT and the multiple peak GCC-PHAT exhibit a reduction in accuracy as SNR decreases. The estimation of τ_2 was above 95% accurate to an SNR of 32dB whereas the estimation of τ_1 was never more than 90% accurate and dropped below 85% accuracy at 46.5dB SNR. This shows that both methods are very sensitive to noise due to the differences in amplitude of the peaks due to distance from the microphones. This is because s_2 was placed closer to m_2 than s_1 was to m_1. The result of this is that the peak caused by s_2 would be of a higher amplitude and would therefore stay above the noise floor to lower SNR, leading to a higher accuracy of delay estimation for that source.

6. CONCLUSIONS AND FUTURE WORK

It has been shown that using a single GCC-PHAT calculation to extract multiple time delays rather than extracting a single delay produces the same results, with a reduction in the number of calculations required per time window. It also provides a rough estimate of the position of the sources with respect to the microphones. The multiple peak GCC-PHAT is also able to calculate delays where the number of sources is greater than or equal to the number of microphones, whereas the traditional single peak GCC-PHAT is unable to as the maximum number

of calculations is less than the number of delays to be calculated.

The proposed method has been shown to be susceptible to additional noise and the accuracy is also dependant on the position of the sources to the microphones. The multiple peak GCC-PHAT could be adapted to increase the accuracy with additional noise using methods such as averaging and accumulation. The method could also be used to estimate the changing delay of a moving source. The robustness of the method to correlated noise, i.e.. reverberation, has also yet to be investigated. The accuracy of the method could also be tested using real recordings in a real space as opposed to the simulations used in this paper.

Research will continue into using multiple delays to reduce the effects of comb filtering in multiple source, multiple microphone configurations in live sound.

7. REFERENCES

[1] J. Chen, J. Benesty, and Y. A. Huang, "Time delay estimation in room acoustic environments: An overview," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1 – 19, 2006.

[2] C. H. Knapp and G. C. Carter, "Generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[3] E. Perez Gonzalez and J. Reiss, "Determination and correction of individual channel time offsets for signals involved in an audio mixture," in *Proceedings of the 125th Audio Engineering Society Convention*, (San Francisco, USA), 2008.

[4] N. Cho and C.-. J. Kuo, "Underdetermined audio source separation from anechoic mixtures with long time delay," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009.

[5] M. Baeck and U. Zölzer, "Real-time implementation of a source separation algorithm," in *Proceedings of the 6th International Conference on Digital Audio Effects (DAFx-03)*, (London, UK), 2003.

[6] Ö. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on Signal Processing*, vol. 52, July 2004.

[7] J. Meyer, "Precision transfer function measurements using program material as the excitation signal," in *Proceedings of the 11th International Conference of the Audio Engineering Society: Test and Measurement*, (Portland, Oregon), 1992.