FAST ACCURATE TIME DELAY ESTIMATION BASED ON ENHANCED ACCUMULATED CROSS-POWER SPECTRUM PHASE

Radu-Sebastian Marinescu[12], Andi Buzo[2], Horia Cucu[2], Corneliu Burileanu[2]

[1]Rohde & Schwarz
[2]Univeristy POLITEHNICA of Bucharest, Romania
radu-sebastian.marinescu@rohde-schwarz.com, {andi.buzo, horia.cucu}@upb.ro, cburileanu@messnet.pub.ro

ABSTRACT

The problem of time delay estimation (TDE) has many approaches and a large field of applications, making it an important research issue. For specific air traffic control systems, time delay estimation is a primary step for speech enhancement. In this paper we introduce two new TDE methods, based on cross power-spectrum phase (CSP), combining previous efficient methods that use accumulating cross power spectrum, whitening and coherence. We show that the proposed methods bring an accuracy improvement of more than 5%, while being 5% to 20% faster.

Index Terms — Time Delay Estimation, Cross Spectrum Phase, Generalized Cross-Correlation

1. INTRODUCTION

Echo canceling, acoustics, radar and sonar localization, seismic and medical processing, pattern detection and speech enhancement are only a few fields where time delay estimation (TDE) remains an active part and had been a hot topic for years.

One of the particular applications in which a TDE block is mandatory is airline and naval traffic control. In this field of application the aim is to assure a good communication between the pilot and the traffic operator. In the typical scenario there are several radio stations on the ground, which receive the pilot's transmission. All these radio stations transmit the pilot's message further to the command center through a VoIP network. This communication channel is not only noisy, but it is also a source of random delay for every new transmitted message. Consequently an accurate TDE block is mandatory for aligning the various received messages in order to make them usable for any subsequent speech enhancement block. Moreover, this TDE block must be characterized by a short response time, because the maximum allowed delay between the pilot's emission and the message reception (in the command center) is 300ms.

The various approaches to time delay estimation can be grouped in three main categories: a) generalized cross-correlation (GCC), b) least mean square (LMS) adaptive filtering and c) adaptive eigenvalue decomposition (EVD). The latter of these three approaches (EVD) was shown to be very efficient in reverberant environments [1]. Despite the variety of optimized adaptive filtering methods (second approach), which have a high accuracy, all of them need a long adaption time, of at least 1 second [2-8]. This makes them unusable for real-time systems, which need quick response (e.g. less than 300ms in traffic control).

The generalized cross-correlation method was introduced back in 1976 by Knapp and Carter [9]. Besides the basic GCC function, they also propose a particular GCC weighting function called Cps-m. Starting from this work, several variations of the GCC weighting function were proposed over the years: ROTH and SCOT [10], Eckart [11], Phase Transform (PHAT) or Cross-power Spectrum Phase (CSP) [12][13], Wiener [14], HT (ML) [15], HB [16]. In [17], all these methods were reviewed and compared based on the mean value and root mean square deviation of the estimated delays. A newer and improved GCC weighting function (*ρ-CSPC*) was recently described by Shean and Liu in [18]. This approach was shown to be much more accurate than the *Cross-power Spectrum Phase (CSP)* in an acoustic source localization scenario because it adds a whitening parameter to discard the non-speech amount on low frequencies and minimum of the coherence function to reduce errors for relatively small energy signals. Prior to this, an innovative *accumulation principle* for computing the GCC was proposed by Matassoni and Svaizer in [19]. In this work, the authors show that their method (*acc-CSP*) outperforms the popular CSP in terms of computational load, while preserving almost the same estimation accuracy. This is possible because the method proposes only one Inverse Discrete Fourier Transform, no matter how many accumulated frames are being used.

The work presented in this study takes advantage of the higher accuracy which characterizes the *ρ-CSPC* method [18] and the robustness and lower computational load of the *accumulated CSP* (*acc-CSP*) method [19]. We propose two new GCC-based estimation methods, by introducing the whitening parameter and minimum of the coherence function from *ρ-CSPC,* in *acc-CSP* method. We compare

our approaches with the previous methods on the Noizeus database [20] and show that they are both faster and more accurate.

The paper is organized as follows. In first part of section 2 we present the TDE background and CSP based method, proposing in second part our new *acc-ρCSP* and *acc-ρCSPC* methods. Experimental results and discussion are provided in section 3. Finally, conclusions and further work stand on section 4.

2. THE PROPOSED TDE METHODS

2.1. Background

Time delay estimation aims at finding the relative delay between two signals $y_1(t)$ and $y_2(t)$, which are two noisy and delayed versions of the same transmitted signal $x(t)$. Among the various approaches to TDE, the most popular and time-efficient method family seems to be the one based on the cross-correlation of the two signals. The so-called generalized cross-correlation was introduced in [9] and it incorporates a filtering function:

$$R^{g}_{y_1y_2}(t) = \int_{-\infty}^{\infty} \Psi(f) \cdot G_{y_1y_2}(f) \cdot e^{j2\pi ft} df \qquad (1)$$

where Gy_1y_2 represents the cross-power spectrum and $\Psi(f)$ is a general frequency weighting with the purpose to facilitate the delay estimation, by emphasizing some spectral information that can depend on source and noise characteristics [13]. Thus the delay is the value of t that maximizes the cross-correlation function.

A derivation of the GCC is PHAT or CSP. Contrary to other techniques, CSP does not imply previous knowledge of the source signal and noise. This approach is independent of the input waveform characteristics, unless signals are strictly narrowband [13]. It was applied in various scenarios and it was shown to be very efficient for time delay estimation [12][13][19]. In CSP the weighting function Ψ_g is computed as follows:

$$\Psi(f) = 1/\left|G_{y_1y_2}(f)\right| \qquad (2)$$

In TDE it is common to split the analysis window into frames of smaller sizes. This is useful especially for moving signal sources (which is the case for traffic air control applications). The TDE is made based on the results obtained from all frames. Because the response time is crucial in some applications, two factors require particular attention: (1) the window length and (2) processing time. A larger analysis window helps to increase the ability of a correct estimation. On the other hand, a larger window means a longer waiting time for filling the buffer. The number of frames within the window directly affects the processing time. The greater the number of frames, the greater the number of required Inverse Discrete Fourier Transforms (IDFT) will be (equation 1).

Under these circumstances, an alternative way to use many analysis frames, while preserving a small computation time, is the *accCSP method*, proposed in [19]. This method estimates the CSP by averaging the CSP over all frames. In frequency domain this corresponds to an averaged cross-power spectrum:

$$G_{CSP}(f) = \sum_{k=1}^{K} \frac{G_{y_1y_2,k}(f)}{\left|G_{y_1y_2,k}(f)\right|}, \qquad (3)$$

where K represents the number of accumulated frames.

The *acc-CSP* method has two advantages. First *the computational complexity is reduced*, because the number of IDFTs drops to only one. Second, the intrinsic integration effect contributes to *enhance the estimation*, provided that the delay is fixed during the time interval corresponding to the analysis frames.

A significant improvement of the CSP method is proposed recently in [18]. It modifies the weighting function $\Psi(f)$ as shown in equation (4):

$$\Psi_{y_1y_2}(f) = \frac{1}{\left|G_{y_1y_2}(f)\right|^{\rho} + \min\left[\gamma^2_{y_1y_2}(f)\right]} \qquad (4)$$

where $\gamma^2_{y_1y_2}(f)$ is the signal's coherence function:

$$\gamma^2_{y_1y_2}(f) = \frac{\left|G_{y_1y_2}(f)\right|^2}{G_{y_1}(f) \cdot G_{y_2}(f)} \qquad (5)$$

The tuning parameter ρ (with values between 0 and 1) is a whitening parameter, which discards the non-speech portion of the CSP (below 200Hz) [18][21]. In addition, the presence of the minimum of the coherence function reduces errors for relatively small energy signals, in equation (5).

2.2 Improving the accuracy of acc-CSP method

Taking advantage of both *accCSP* and *ρ-CSPC* methods, we propose using a new *accumulated ρ-Cross Power Spectrum Phase with Coherence* (*acc-ρCSPC*), defined as follows:

$$G_{acc-\rho CSPC}(f) = \sum_{k=1}^{K} \frac{G_{y_1y_2,k}(f)}{\left|G_{y_1y_2,k}(f)\right|^{\rho} + \min\left[\gamma^2_{y_1y_2,k}(f)\right]} \qquad (6)$$

This approach benefits from the strong points of the previous methods (speed and accuracy) and proves to have a better accuracy even for very low signal-to-noise ratios (SNR). The whitening parameter (ρ) is used to emphasize the speech regions in the spectrum (provided that SNR > 0dB) and consequently reducing the impact of noise outside the speech regions. The minimum coherence function added in the formula limits the effect of the denominator for signal segments with small energy. The accumulating scheme leads to a faster version of the previous non-accumulating algorithms, with the possibility of using smaller frame sizes to provide better results in unfavorable conditions.

For scenarios without relatively small energy signals and fast response time demanding, we propose a new method *accumulated ρ-Cross Power Spectrum Phase* (*acc-ρCSP*), eliminating the minimum coherence function:

$$G_{acc-\rho CSP}(f) = \sum_{k=1}^{K} \frac{G_{y_1y_2,k}(f)}{\left|G_{y_1y_2,k}(f)\right|^{\rho}} \qquad (7)$$

This way we can achieve an even faster computing time compared with the previous *acc-ρCSPC* method, because now we do not need to calculate the minimum coherence for every frame. The experimental results proved that by removing the minimum coherence term, the accuracy is not affected for ρ lower than a threshold.

3. EXPERIMENTAL RESULTS AND DISCUSSIONS

3.1. Experimental setup

In order to evaluate the improvement brought by the new proposed *acc-ρCSPC* and *acc-ρCSP* methods we have also implemented the starting methods *accCSP* and *ρCSPC*, in Matlab. The analyzed signals were selected from the Noizeus database [20], which is a noisy speech corpus used for evaluation of speech enhancement algorithms. The noisy database contains 30 sentences (produced by three male and three female speakers at 8 KHz) corrupted by 8 different real-world noises at 4 different SNRs. The noise of the Noizeus sentences was taken from the AURORA database [22] and includes suburban train noise, babble, car, exhibition hall, restaurant, street, airport and train-station noise.

To avoid spurious results we divided the Noizeus database in two parts: 50% of the sentences were used for *development* and the other 50% were used for *evaluation* (15 sentences for each part). The signal pairs chosen for alignment cover all the combinations of noise types: $C_2^8 = 28$. By using 4 different SNR levels and 5 artificially introduced delay values (5, 10, 25, 50 and 100 ms), the total number of test pairs becomes 28x15x4x5=8400. In the development stage we varied the following parameters: SNR, frame sizes, overlap factor and number of averaged frames. This stage also aims at finding a proper value for the whitening factor ρ. In this stage we evaluate both *acc-ρCSP* and *acc-ρCSPC*.

The metrics used for evaluation are the average relative error (equation 8) and the standard deviation of the relative error.

$$\widetilde{\Delta}_{|\varepsilon|} = \left(\sum_{i=1}^{N} |\varepsilon_i| \right) / N \qquad (8)$$

where ε_i represent the relative error of the estimated delay for the i^{th} signal pair and N is the total number of analyzed signal pairs.

The other metric used is accuracy, defined as the ratio between the number of correctly estimated delays and the total number of estimations performed (we understand a correct estimation as one where the difference between estimated and introduced delay is 0 samples). In the particular scenario investigated in this work, the accuracy is the most important metric because the TDE block must provide a perfect alignment. This condition is imposed by the subsequent speech enhancement blocks which cannot work on delayed signals.

3.2. The calibration stage

In this stage we have used only the development data set. In order to obtain efficient results, it is important to calibrate adequately the method's parameters, which influence the accuracy of the *acc-ρCSPC* method. The number of frames, the number of samples/frame and overlap factor are chosen based on the nature of the application by making a trade-off between computing time, accuracy and fast response of the system. In our application, we chose $K = 4$ averaging frames of 1024 samples each, and an overlap factor of 25%.

Instead, the ρ parameter requires particular attention because it characterizes the proposed methods. Thus, Figure 1 confirms that there is an optimal value for ρ. This value depends on SNR. For higher SNRs, the optimal ρ has a greater value. For the comparison with other methods we have chosen $\rho = 0.73$ as this is the value that maximizes the average accuracy.

Figure 2 shows the effect of the omitted coherence term in formula (7) from the accuracy point of view. This term makes the difference between *acc-ρCSPC* and *acc-ρCSP*. It can be observed for $\rho \in [0, 0.77]$ the accuracy of the two methods is equal. For $\rho > 0.77$ *acc-ρCSPC* outperforms *acc-ρCSP*. However, these are not usual values for ρ (we maximized the accuracy with ρ =0.73). Hence, in order to improve the computation speed the *acc-ρCSP* method can be chosen instead of *acc-ρCSPC*.

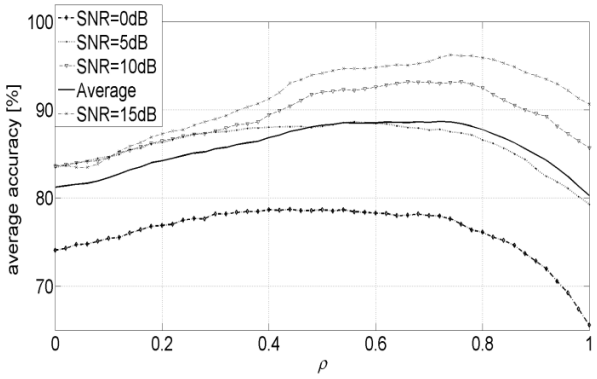

Figure 1. The influence of SNR and ρ over the *acc-ρCSPC* accuracy

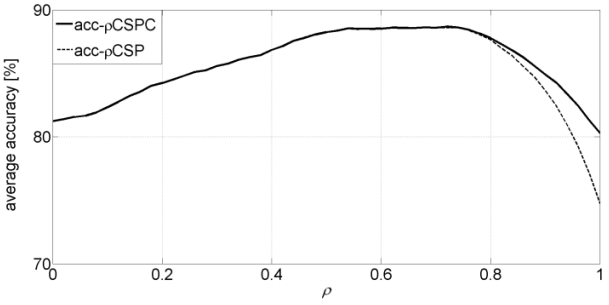

Figure 2. The dependence of *acc-ρCSP* and *acc-ρCSPC* on ρ

After calibrating the *acc-pCSPC* method on the development database, we further compare it with the previous approaches to TDE (*acc-CSP* [19] and *pCSPC* [18]) on the evaluation database.

3.3. The comparison with the previous methods

The evaluation conditions are chosen to satisfy the real-time requirements from the air traffic control scenario. The values for frame size and overlap factor were fixed to 1024 samples, respectively 25%. The values used for SNR and delay are the ones described in Section 3.1.

The obtained results are shown in Table I. For *acc-ρCSPC* and *acc-CSP* the results are available only in the case of 4 and 8 accumulating frames. For a single frame we cannot talk about any accumulation of frames; in this case the *acc-CSP* and the *acc-ρCSPC* are equivalent with the CSP and the *ρCSPC* methods respectively. On the other hand, the accuracy for *pCSPC* is available only for a frame computation, because the method does not consider any accumulating scheme.

The results show that the proposed *acc-ρCSPC* outperforms the other previous methods by obtaining higher accuracy and a lower relative error in delay estimation. For 8 accumulating frames it improves the accuracy of *acc-CSP* by an absolute value of 3.3 %, while for 4 accumulating frames (which is the common case for air traffic control applications) the absolute accuracy improvement is 6.7%. In addition, the lower values for average relative error and standard deviation of the relative error show that even if the signals are not perfectly aligned the gap is smaller than in the other methods. This means that *acc-ρCSPC* is more efficient even for applications where the perfect alignment is not compulsory, for example signal spotting.

The dependence of the computation time on the frame/ window size is presented in Table II. The results are obtained from the C language implementation of the *acc-ρCSPC* method. The algorithm ran on a machine with Intel i5 processor, 4 GB RAM and Ubuntu 12.10 as the operating system. We have chosen different window sizes for evaluation. For each window size we have chosen the possible values for the frame size. For example, if the window size is 1024 samples, the possible sizes for the frame are 512 samples and 1024 samples. The N/A cells indicate the case where the frame size is larger than the window size. It must be noted that the frame size cannot be too small because the Fourier Transform (FT) would not make any sense. The number of samples per frame is chosen to be a power of 2 in order to facilitate the fast computation of the FT.

Table I. Accuracy evaluation for the TDE methods

	Method \ No.Frames	1	4	8
Accuracy (%)	*acc-CSP*	N/A	91.1	96.3
	ρCSPC	79.9	N/A	N/A
	acc-ρCSPC	N/A	97.8	99.9
$\widetilde{\Delta}_{\lvert\varepsilon\rvert}$	*acc-CSP*	N/A	45.1	8.02
	ρCSPC	215	N/A	N/A
	acc-ρCSPC	N/A	5.91	0.17
STD	*acc-CSP*	N/A	141	23.5
	ρCSPC	217	N/A	N/A
	acc-ρCSPC	N/A	18.6	0.89

Table II. Processing time dependence on window/frame size

Frame size	Computing time in µs for windows of			
	1024 samples	2048 samples	4096 samples	8192 samples
512	151	289	565	1058
1024	*163*	302	582	1081
2048	N/A	*330*	617	1124
4096	N/A	N/A	*670*	1193
8192	N/A	N/A	N/A	*1305*

The results show that by increasing the number of frames within the analysis window the processing time is reduced. For example, for a long window size of 8192 samples which is divided in 2 smaller frames of 4096 samples the processing time decreased with almost 8.6%, from 1305 µs to 1193 µs. If we divide the window in 4 frames of 2048 samples the processing time decreases even more, with almost 14%, from 1305 µs to 1124 µs. We stopped with iterations at 16 frames of 512 samples for which the processing time decreased with 19%, from 1305 µs to 1058 µs. This is explained by the complexity of the IDFT calculated as Fast Fourier Transform (FFT). For a frame size of N samples the FFT complexity is $O(N \cdot \log N)$. In *acc-ρCSPC*, the weighting function is calculated as an average over K frames of N/K samples each. Hence, the FFT complexity in this case is $O(N \cdot \log(N/K))$.

The results also suggest that longer windows require longer processing time. On the other hand, longer windows provide better accuracy. Depending on the time and accuracy constrains that each application imposes the appropriate configuration may be chosen.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we presented two new methods (*acc-ρCSP* and *acc-ρCSPC*) based on the combination of two cross-power spectrum based methods. The previous methods are enhanced by applying the accumulating principle and by

introducing a whitening factor. The proposed methods improve the accuracy and the computation speed of TDE. The experiments were run on the standard *Noizeus* database. The evaluation was made based on three metrics: accuracy, average relative error and standard deviation of the relative error. The experimental results proved that the proposed methods outperform the previous ones by all metrics.

The enhanced approach is a real solution for realigning noisy signals in applications such as echo canceling, radar and sonar localization, seismic and medical processing, pattern detection and speech enhancement, providing a higher TDE accuracy and faster processing time by 5 – 20% than the previous GCC methods.

Future work will involve implementation of *acc-ρCSPC* and acc-*ρCSP* in VoIP environment applications. Methods characterization for different developed system will also involve a specific analysis.

5. REFERENCES

[1] J. Benesty, "Adaptive eigenvalue decomposition algorithm for passive acoustic source localization", *J. Acoust. Soc. Am. Volume 107, Issue 1,* pp. 384-391, 2000.

[2] B. Widrow, S.D. Streams, "Adaptive signal processing", Penitence-Hall, ISBN 0130040290, USA, 1985

[3] S.N. Lin and S.J. Chern, "A new adaptive constrained LMS time delay estimation algorithm", *Signal Processing, Volume 71, Issue 1*, pp. 29-44, November 1998.

[4] A.W.H. Khong and P.A. Naylor, "Efficient Use Of Sparse Adaptive Filters", *In Proceedings of ACSSC '06, Fortieth Asilomar Conference on Signals, Systems and Computers*, article ID 10.1109/ACSSC.2006.354982, pp. 1375-1379, 2006.

[5] R.A. Dyba, "Parallel Structures for Fast Estimation of Echo Path Pure Delay and Their Applications to Sparse Echo Cancellers", *In Proceedings of CISS 2008, 42nd Annual Conference on Information Sciences and Systems*, article ID 10.1109/CISS.2008.4558529, pp. 241-245, 2008.

[6] D. Hongyang and R.A. Dyba, "Efficient Partial Update Algorithm Based on Coefficient Block for Sparse Impulse Response Identification", *In Proceedings of CISS 2008, 42nd Annual Conference on Information Sciences and Systems*, article ID 10.1109/CISS.2008.4558527, pp. 233-236, 2008.

[7] D. Hongyang and R.A. Dyba, "Partial Update PNLMS Algorithm for Network Echo Cancellation*", In Proceedings of ICASSP 2009, IEEE International Conference on Acoustics, Speech and Signal Processing*, article ID 10.1109/ ICASSP.2009.4959837, pp. 1329-1332, 2009.

[8] K. Sakhnov, E. Verteletskaya, and B. Simak, "Partial Update Algorithms and Echo Delay Estimation," Communications – *Scientific Journal of the University of Zilina*, Zilina – Slovakia, vol. 13, no. 2, p. 14-19, 2011.

[9] C. Knapp and G.C. Carter, "The Generalized Correlation Method for Estimation of Time Delay", *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, issue 4, pp. 320-327, 1976.

[10] D.H. Youn, N. Ahmed, and G.C. Carter, "On the Roth and SCOTH Algorithms: Time-Domain Implementations*", In Proceedings of the IEEE, vol. 71, issue 4*, pp. 536-538, 1983.

[11] Q. Tianshuang and W. Hongyu, "An Eckart-weighted adaptive time delay estimation method*", IEEE Transactions on Signal Processing, vol. 44, issue 9*, pp. 2332-2335, 1996.

[12] M. Omologo and P. Svaizer, "Acoustic event localization using a crosspower-spectrum phase based technique", *Proceedings of ICASSP*, Australia, pp. 273-276, 1994.

[13] M. Omologo and P.Svaizer, "Use of the crosspower-spectrum phase in acoustic event location", *IEEE Transactions on Speech Audio Process*, pp. 288-292, 1997.

[14] V. Zetterberg, M.I. Pettersson, and I. Claesson, "Comparison Between Whitened Generalized Crosscorrelation and Adaptive Filter for Time Delay Estimation*", In Proceedings of TS/IEEE, OCEANS, vol. 3*, article ID 10.1109/OCEANS.2005.1640117, pp. 2356 – 2361, 2005.

[15] K.W. Wilson and T. Darrell, "Learning a Precedence Effect-Like Weighting Function for the Generalized Cross-Correlation Framework", *IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, issue 6*, pp. 2156-2164, 2006.

[16] Y. Sun and T. Qiu, "The SCOT Weighted Adaptive Time Delay Estimation Algorithm Based on Minimum Dispersion Criterion*", In Proceedings of the ICICIP Conference on Intelligent Control and Information Processing*, pp. 35-38, 2010.

[17] K. Sakhnov, E. Verteletskaya, and B. Simak, "Echo Delay Estimation Using Algorithms Based on Cross-correlation," *Journal of Convergence Information Technology*, Volume 6, Number 4, pp. 1 – 11, April 2011.

[18] M. Shean and H. Liu, "A Modified Cross Power-Spectrum Phase Method Based on Microphone Array for Acoustic Source Localization," *IEEE International Conference on System, Man and Cybernetics*, San Antonio, TX, USA, pp. 1286 – 1291, 2009.

[19] M. Matassoni and P. Svaizer, "Efficient Time Delay Estimation Based on Cross-Power Spectrum Phase", *European Signal Processing Conference (EUSIPCO)*, Florence - Italy, 2006.

[20] NOIZEUS: A noisy speech corpus http://www.utdallas.edu/ ~loizou/speech/noizeus/ [retrieved: Feb, 2013]

[21] D.V. Rabinkin, R.J. Renomeron, A. Dahl, J.C. French, J.L. Flanagan, and M.H. Bianchi, "A DSP Implementation of Source Location Using Microphone Arrays", *The Journal of the Acoustical Society of America, Volume 99, Issue 4*, pp. 2510-2527, April 1996.

[22] AURORA database, http://www.elda.org/article52.html [retrieved: Feb, 2013]