Real-Time DSP Implementation of a Subband Beamforming Algorithm for Dual Microphone Speech Enhancement

Zohra Yermeche, Benny Sällberg, Nedelko Grbić and Ingvar Claesson
Blekinge Institute of Technology, School of Engineering
SE-372 25 Ronneby, Sweden
Email: zye@bth.se, bsa@bth.se, ngr@bth.se, icl@bth.se

Abstract—A real-time Digital Signal Processor (DSP) based implementation of a subband beamforming algorithm and its evaluation for dual microphone speech enhancement is presented. The algorithm, a calibrated constrained beamformer, is described theoretically and a real-time structure is proposed, including an efficient approach for multichannel data transformation. Measurements show that the battery driven DSP implementation supports 20 h operation-time, with an improved Signal-to-Noise Ratio (SNR) of up to 14 dB in high-noise factory environment. Further, less than half the provided computational performance of the DSP is used by the proposed method, hence, processing of additional tasks may be included.

I. INTRODUCTION

Speech enhancement strategies are crucial for speech acquisition applications in adverse noise environments. In these applications, the objective is to enable natural speech communication for speakers at remote distances from acquisition devices. The robustness of speech enhancement methods is a prerequisite to their implementation in the DSP environment and for real-world applications. Furthermore, the low computational complexity of such algorithms constitutes a major requirement for making real-time processing feasible. A hybrid (analog-digital) implementation of a low complexity algorithm for speech enhancement, the Adaptive Gain Equalizer (AGE), was presented and evaluated in [1]. However, common for single microphone approaches such as the AGE, is the degrading performance in low SNRs, while spatial information is not available.

Methods using microphone arrays enabled the development of combined temporal and spatial filtering algorithms known as beamforming techniques [2]. Many of the beamforming techniques rely on Voice Activity Detection (VAD), needed in order to avoid source signal cancellation effects, which may result in unacceptable levels of speech distortion [2]. Calibration-based methods were developed to circumvent the need of a VAD as well as to take into consideration the real environment’s acoustical properties [3], [4]. An adaptive calibrated subband beamforming algorithm, known as the Calibrated Weighted Recursive Least Squares (CWRLS) beamformer, was assessed theoretically and practically in previous studies [4], [5]. A real-time implementation of the CWRLS beamformer on a floating point DSP, using an efficient transform approach for the processing of three-dimensional data, is described in this paper. The acoustic noise suppression, perceptual speech quality and computational performance evaluation of the proposed solution is presented.

In situations where the worker uses devices, such as hearing protection headsets, in very noisy environments (for protection, safety and comfort), it is desired that the devices’ operation-time is at least one full working day (eight hours in Sweden). Hence, the battery life-time is of vital importance in such mobile applications. A current consumption measurement for the developed DSP-based speech enhancement solution is conducted with the intention to predict the expected battery life-time of the proposed implementation.

II. CONSTRAINED SUBBAND BEAMFORMING

In subband beamforming, the multichannel filtering operations are performed on the array input signals for each subband frame independently, as depicted in Fig. 1. Subband processing results in a computational gain, since the filtering of narrow band signals requires lower sample rates. Hence, in an efficient implementation, the spectral decomposition of the inbound signals is followed by a decimation operation [6].

A sound field generated by a desired source $s(t)$ and interfering noise $n(t)$, at time instant $t$, is impinging on an array of $M$ sensors. A multichannel analysis filter bank is applied to the inbound signals. The noisy signal received by the $m^{th}$ sensor, $x_m(t) = s_m(t) + n_m(t)$, $(m = 1, 2, \ldots, M)$, is sampled and decomposed into a set of $K$ narrow band signals, $X_m^k(l) = S_m^k(l) + N_m^k(l)$, where $k = 0, 1, \ldots, K - 1$, is the subband index, and $l$ is the sample index after decimation by a factor $D$.

To avoid aliasing effects, a two-times oversampled structure is used (i.e., $D = \frac{N}{2}$). The spatial filtering is performed on the array input vector, $X^k(l) = [X_1^k(l), X_2^k(l), \ldots, X_M^k(l)]^T$, for each subband individually. Hence, the beamformer’s output for subband $k$, $Y^k(l) = W_k^H X^k(l)$, where $W_k$ is the beamformer weight vector. The symbols $()^T$ and $()^H$ stand for the transpose and Hermitian transpose, respectively. The subband outputs of the subsequent filters are combined by a synthesis filter bank, in order to create a time-domain output signal, as illustrated in Fig. 1.

The constrained subband beamformer considered here is deduced from an RLS formulation of the Wiener solution [5], with the array weight vector given by
Subband beamforming structure.

Here, \( \hat{R}_s^{[k]}(l) \) is the received signal covariance matrix estimate, continuously calculated from observed data by

\[
\hat{R}_s^{[k]}(l) = \sum_{p=0}^{l-1} \lambda^{-p-1} X^{[k]}(p) X^{[k]}(p)^H ,
\]

where \( \lambda \) is a forgetting factor used to track variations in the surrounding noise environment. The spatial source covariance matrix, \( R_s^{[k]} = E\{S^{[k]}(l) S^{[k]}(l)^H\} \), and the spatial cross covariance vector, \( r_s^{[k]} = E\{S^{[k]}(l) S^{[k]}(l)^*\} \), are defined for \( S^{[k]}(l) = [S_1^{[k]}(l), S_2^{[k]}(l), \ldots, S_M^{[k]}(l)]^T \) (i.e., the source only received input vector) and the desired signal, \( S^{[k]}(l) \). The operator \( E\{\cdot\} \) is the statistical expectation and \( (\cdot)^* \) is the complex conjugate. The source covariance matrix constitutes a soft constraint which acts as a spatial passband. This moderates the weight fluctuations generated by speech pauses, and also, forces full rank properties of the total matrix, \( \hat{R}_s^{[k]}(l) = \hat{R}_s^{[k]}(l) + R_s^{[k]} \). The adaptive solution of the least-squares optimization problem given in (1) is based on a prerequisite knowledge of the source statistics, \( R_s^{[k]} \) and \( r_s^{[k]} \). These statistics can be estimated based on some knowledge about the source position. In various hands-free communication applications, the source position is almost fixed and known approximately. Hence, source statistics can be estimated from data sequences uttered from the known source position and gathered by the array in a calibration phase.

III. REAL-TIME STRUCTURE OF THE CWRLS BEAMFORMER

A recursive formulation for the update of the beamforming weight vector was derived in [4]. A real-time implementation of the resulting CWRLS beamformer is illustrated in Fig. 2, and each part is described in the following subsections:

A. Subband Decomposition

The sound field impinging on the sensor array is sampled and the resulting signals are decomposed into their spectral components. This later operation is performed by a two-times oversampled multichannel modulated uniform analysis filter bank structure, greatly simplified through the use of a polyphase implementation [6]. The prototype filter is a Hamming window of length \( 3K \).

Fig. 2. Block diagram of the CWRLS beamformer’s real-time implementation.

B. Calibration Phase

In an initial calibration phase, prior to the filtering operations, a calibration sequence is emitted from the target source position and gathered in a quiet environment, in order to estimate the desired source statistics. Since in a realistic scenario the reference source signal, \( S^{[k]}(l) \), is not directly available, the received input signal of a selected sensor \( m' \), \( S_m^{[k]}(l) \), is used instead. Hence, the desired signal consists of the reverberant multi-path source signal received at sensor \( m' \). The benefit is that the real room acoustical properties are taken into consideration, including errors due to microphone mismatch. The calibration phase consists of the following two steps, run for each subband \( k = 0, 1, \ldots, K - 1 \):

- Calculate the source covariance estimates for a data set of \( L \) decimated samples following

\[
\hat{R}_s^{[k]} = \frac{1}{L} \sum_{l=0}^{L-1} S^{[k]}(l) S^{[k]}(l)^H ,
\]

\[
\hat{R}_s^{[k]} = \frac{1}{L} \sum_{l=0}^{L-1} S^{[k]}(l) S_{m'}^{[k]}(l)^* .
\]

- Calculate the eigenvector matrix and the eigenvalue matrix, \( Q_s^{[k]} = [q_1^{[k]}, q_2^{[k]}, \ldots, q_M^{[k]}] \) and \( \Gamma_s^{[k]} = \text{diag}(\gamma_1^{[k]}, \gamma_2^{[k]}, \ldots, \gamma_M^{[k]}) \), respectively, of the source covariance matrix estimate, following

\[
\hat{R}_s^{[k]} = Q_s^{[k]} \Gamma_s^{[k]} Q_s^{[k]H} .
\]
\[
\lambda^{-2} \hat{R}^{[k]-1} (l-1) X^{[k]}(l) X^{[k]}(l)^H \hat{R}^{[k]-1} (l-1) \\
+ \lambda^{-1} X^{[k]}(l)^H \hat{R}^{[k]-1} (l-1) X^{[k]}(l) \\
\]
\[
\hat{R}^{[k]-1} (l) = D - \frac{(1-\lambda) \gamma_s[k] D q_s[p] q_s[p]^H D}{1 + (1-\lambda) \gamma_s[k] q_s[p] H D q_s[p]},
\]
where \(D\) is an intermediate matrix variable, \(p = \lfloor \text{mod} M \rfloor + 1\), and \(\text{mod}\) represents the modulus function.

- Calculate the weight vector, \(W^{[k]}\), following a smoothing first order AR-model with parameter \(\eta\), as
\[
W^{[k]} = \eta W^{[k]}_{[l-1]} + (1-\eta) \hat{R}^{[k]-1} (l) \hat{R}^{[k]},
\]
- Calculate the output signal \(Y^{[k]}(l)\) following
\[
Y^{[k]}(l) = W^{[k]}_I X^{[k]}(l).
\]

**D. Subband Reconstruction**

The outputs of the \(K\) subband beamformers form the inputs to a modulated uniform synthesis filter bank, so as to create the time-domain output.

**IV. DSP IMPLEMENTATION**

The implementation of the CWRLS algorithm is made on a floating point DSP named ADSP21262 from Analog Devices. This is a high-performance DSP that supports effective parallel computations through the Single Instruction Multiple Data (SIMD) mode, suitable for vector-based operations. The algorithm is efficiently implemented using a transformed approach to reduce the overhead. A rule-of-thumb for the transformed approach is to perform all computations so that the largest data dimension is used in the element-wise vector operations. The data transformation approach for efficient implementation is described in this section. Further, the eigenvalue decomposition on the DSP, using Given’s rotation, is specified.

**A. Data Transformation**

The presented data transformation for efficient implementations does not alter the data itself. This alternative way of managing the data was introduced in [8] for a Matlab-based real-time implementation. The computational overhead for a vector operation of a long vector is lower than the overhead of several consecutive loop iterations, due to the fact that the source code compiler can parallelize the computations using the SIMD mode of the DSP. The transformation approach focuses on minimizing the number of loop iterations in the software, and instead utilizes longer vector operations. Consider the following primal example where an input data matrix \(X = [X^{[0]}, X^{[1]}, \ldots, X^{[K-1]}]_{M \times K}\), with each \(X^{[k]} = [X^{[k]}_1, X^{[k]}_2, \ldots, X^{[k]}_M]_{M \times 1}\). In this example it is desired to compute a vector outer product of size \(M \times M\) for each subband at a time, according to \(P^{[k]} = X^{[k]} X^{[k]} H\) (this is identified as a typical operation in the RLS structure). This operation would require a total of \(K\) loop iterations to compute all output elements of the resulting three-dimensional matrix \(P\) of size \(M \times M \times K\). In the transformed approach the input data are ordered as, \(X' = X'^T = [X^{[1]}, X^{[2]}, \ldots, X^{[M]}]_{K \times M}\), where each \(X^{[m]} = [X^{[m]}_1, X^{[m]}_2, \ldots, X^{[K-1]}_m]_{K \times 1}\). Consequently, performing a similar outer product computation for each element in the resulting matrix yields \(P_{r,c} = X'^r \odot X'^c\), where \(r\) and \(c\) represent the row and column indexes, respectively, of the output data matrix \(P\), and \(\odot\) represents an element-wise multiplication. In our example, the transformed approach requires a total of \(M^2\) loop iterations, and while \(M^2 < K\), the computational overhead due to loop iterations is significantly reduced. A subband-specific outer product, for \(K = 128\) subbands and \(M = 2\) microphones, would require a total of 15363 operations on the DSP\(^1\), while, the transformed approach would require 2046 operations. The transformed approach requires 7.5 times fewer operations than the straightforward approach for computing the same operations.

**B. Eigenvalue Decomposition**

During the calibration phase of the CWRLS, it is needed to compute the eigenvectors and eigenvalues of the estimated source covariance matrix, \(\hat{R}^{[k]}\). This matrix is Hermitian (\(\hat{R}^{[k]} = \hat{R}^{[k] H}\)), and the off-diagonal entries of \(\hat{R}^{[k]}\) appear as complex conjugated pairs. The eigenvalue decomposition in this paper uses Given’s rotation with a preprocessing step [7]. The preprocessing step ensures that the matrix \(\hat{R}^{[k]}\) is transformed into a real valued triangular matrix, for which Given’s rotation is applied in its standard form. The present implementation requires in total 59597 operations to find the eigenvalues and corresponding eigenvectors for 128 complex valued Hermitian matrices of size \(2 \times 2\). The eigenvalue decomposition in the current DSP implementation is solved for one subband at a time, in order to not violate the strict timing conditions imposed by a real-time implementation.

**V. PERFORMANCE EVALUATION**

Simulations were performed with a two microphone setup positioned in the center of a \(3 \times 4 \times 3\) m room. A loudspeaker emitting clean speech sequences from the TIMIT database was positioned at 50 cm from the array center, with an angle of 45 degrees to the array. To simulate ambient noise, four loudspeakers were positioned at the corners of the room. The emitted noise was prerecorded in a factory environment. The DSP hardware structure was utilized with a sampling frequency of 48 kHz. The number of subbands was set to \(K = 256\) and the data block size was set to \(D = 128\). A short-time sequence of a noisy speech input to a single-microphone, with \(\text{SNR} = -10\) dB, and the corresponding CWRLS beamformer’s output are shown in Fig. 3.

The signal bandwidth (i.e., the number of consecutive subbands processed) is varied in Fig. 4. It can be seen that this parameter allows for a trade off between speech enhancement performance versus operation-time and additional processing capabilities. The SNR improvement is given in plot (4a) for different input SNRs. The proposed method enhances the

\(^1\) The analysis of the number of required operations is performed in a simulation environment provided by the DSP manufacturer. The test-software is written in C language, where the compiler is set to operate at the highest optimization level.
speech in low SNR conditions by up to 14 dB. Since the power content of the processed noise and speech signals is mostly situated below 8 kHz, we obtain no noticeable variation in SNR improvement by further increasing the bandwidth.

The perceptual evaluation of the CWRLS beamformer, based on the PESQ P. 862 standard [9], is presented in plot (4b). The two top curves correspond to the PESQ improvement achieved by the proposed solution in comparison to the received noisy signal with different SNRs. The lower (negative-valued) curve indicates the small distortion of clean speech introduced by the beamformer, relative to the unprocessed case (“transparent mode”). The differential PESQ is plotted as a difference in MOS, which is a scale of 1 to 5 to measure speech quality (with 1 for poor quality and 5 for good quality). The results clearly indicate a positive gain in PESQ MOS (up to 0.7 MOS) for the processing by the beamformer of highly corrupted speech, while, low additional distortion (smaller than 0.1 MOS) is caused to the (processed) clean speech.

Expectedly, the current consumption presented in plot (4c) increases with increasing bandwidth. However, in the actual implementation, it does not exceed 250 mA, which can be used as an approximative upper bound of the current consumption. Many modern batteries support up to, and in some cases exceeding, 2.5 Ampere hours [Ah]. In this implementation two batteries are used, hence, the predicted minimum operation-time of the proposed implementation is 20 hours.

A measure of the adaptive algorithm (excluding the calibration phase) computational load is given in plot (4d), as a function of the bandwidth used. The computational load is defined as the measured amount of time the processor performs computations over the total processing time. If, for example, the computational load is 100 % the processor does not support additional processing. The measurement indicates that the computational load of the proposed algorithm never exceeds 50 %. Henceforth, the DSP implementation of the speech enhancement solution presented in this paper provides a considerable computational margin for integrating additional processing of the input (or processed) signal, for instance, speech coding or speech recognition.

VI. CONCLUSION

This paper presents an implementation of a calibrated constrained subband beamformer for speech enhancement. The evaluated performance using the specific hardware matches the performance of previous offline simulations. This contribution shows that the implementation of a calibrated beamformer, such as the CWRLS, is feasible on a DSP in real-time. Furthermore, the predicted operating time of the presented hardware is 20 hours, and the computational load of the method allows for additional processing. Future research on the proposed real-time DSP solution will include structures to compensate for movements of the desired source.

REFERENCES